(2)

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

Automated Performance Evaluation Technique
for Cryptologic Sites

by

John Michael Mackin

September 1992

Thesis Advisor: Donald V. Z. Wadsworth

Approved for public release; distribution is unlimited.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

Form Approved OMB No 0704-0188

1a REPORT SECURITY CLASSIFICATION UNCLASSIFIED	1b RESTRICTIVE MARKINGS
2a SECURITY CLASSIFICATION AUTHORITY	3 DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release; distribution is unlimited.
2b DECLASSIFICATION/DOWNGRADING SCHEDULE	
4 PERFORMING ORGANIZATION REPORT NUMBER(S)	5 MONITORING ORGANIZATION REPORT NUMBER(S)
6a NAME OF PERFORMING ORGANIZATION Naval Postgraduate School; 6b OFFICE SYMBOL (If applicable)	7a NAME OF MONITORING ORGANIZATION Naval Postgraduate School
6c ADDRESS (City, State, and ZIP Code) Monterey, CA 93943-5000	7b ADDRESS (City, State, and ZIP Code) Monterey, CA 93943-5000
8a NAME OF FUNDING/SPONSORING ORGANIZATION; 8b OFFICE SYMBOL (If applicable)	9 PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
8c ADDRESS (City, State, and ZIP Code)	10 SOURCE OF FUNDING NUMBERS: PROGRAM ELEMENT NO; PROJECT NO; TASK NO; WORK UNIT ACCESSION NO

11 TITLE (Include Security Classification)

AUTOMATED PERFORMANCE EVALUATION TECHNIQUE FOR CRYPTOLOGIC SITES

12 PERSONAL AUTHOR(S)

MACKIN, John M.

13a TYPE OF REPORT	13b TIME COVERED	14 DATE OF REPORT (Year, Month, Day)	15 PAGE COUNT
Master's Thesis	FROM ____ TO ____	1992 September	123

16 SUPPLEMENTARY NOTATION The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or U.S. government.

17 COSATI CODES			18 SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB-GROUP	Cryptologic, SOI, RFD, Noise CDAA Performance Evaluation

19 ABSTRACT (Continue on reverse if necessary and identify by block number)

Currently, Naval Security Group (NSG) personnel lack an automated system for evaluating Signals-of-interest (SOI) collection performance of NSG sites. The performance evaluation technique (PET), developed by faculty and students of the Naval Postgraduate School, is intended to meet this need. A means for automating the PET has been developed under this thesis research. This "MATLAB Automated PET System" (MAPS) is described in detail in this thesis and compared with the previous manual PET and a semi-automated version based on GRAFTOOL software. MAPS, based on the high-level language, MATLAB, utilizes measured signal and noise levels, and system gains and losses to evaluate site performance in terms of percent of SOI lost. This information is critical to managers and operators of the various NSG collection sites located throughout the world. It permits managers to assess operator performance, evaluate the impact of encroachments in the vicinity of the site, determine the utility of proposed interference mitigation actions, recognize the effect of natural phenomena (such as

20 DISTRIBUTION/AVAILABILITY OF ABSTRACT ☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT ☐ DTIC USERS	21 ABSTRACT SECURITY CLASSIFICATION UNCLASSIFIED
22a NAME OF RESPONSIBLE INDIVIDUAL WADSWORTH, Donald V.Z.	22b TELEPHONE (Include Area Code) 408-646-2115; 22c OFFICE SYMBOL EC/Wd

DD Form 1473, JUN 86 Previous editions are obsolete. SECURITY CLASSIFICATION OF THIS PAGE

S/N 0102-LF-014-6603 UNCLASSIFIED

BLOCK 19 (Cont.)
solar storms) on SOI collection capability, and predict future SOI collection performance. The manual, semi-automated, and automated PET systems were compared in the areas of cost, speed, ease-of-operation, and accuracy of the performance estimation. MAPS was determined to be the most useful approach for providing the automated PET capabilities to the NSG sites.

Approved for public release; distribution is unlimited.

Automated Performance Evaluation Technique for Cryptologic Sites

by

John Michael Mackin
Lieutenant, United States Navy
B.S., Villanova University, 1985

Submitted in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE IN SYSTEMS ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
September 1992

Author: John M. Mackin
John Michael Mackin

Approved By:
Donald V. Z. Wadsworth, Thesis Advisor

Richard W. Adler, Second Reader

Jeffrey B. Knorr, Chairman
Electronic Warfare Academic Group

ABSTRACT

Currently, Naval Security Group (NSG) personnel lack an automated system for evaluating Signals-of-Interest (SOI) collection performance of NSG sites. The performance evaluation technique (PET), developed by faculty and students of the Naval Postgraduate School, is intended to meet this need. A means for automating the PET has been developed under this thesis research. This "MATLAB Automated PET System" (MAPS) is described in this thesis and compared with the previous manual PET and a semi-automated version based on GRAFTOOL software. MAPS, based on the high-level language, MATLAB, utilizes measured signal and noise levels and system gains and losses to evaluate site performance in terms of percent of SOI lost. This information is critical to managers and operators of the various NSG collection sites located throughout the world. It permits managers to assess operator performance, evaluate the impact of encroachments in the vicinity of the site, determine the utility of proposed interference mitigation actions, recognize the effect of natural phenomena (such as solar storms) on the SOI collection capability, and predict future SOI collection performance. The manual, semi-automated, and automated PET systems were compared in the areas of cost, speed, ease-of-operation, and accuracy of the performance estimation. MAPS was determined to be the most useful approach for providing automated PET capabilities to the NSG sites.

TABLE OF CONTENTS

LIST OF FIGURES

I. INTRODUCTION

The Naval Security Group (NSG) operates and maintains numerous radio-frequency direction finding and signal-collection sites throughout the world. Each of these sites is unique in configuration and in mission; however, most are tasked for the collection of signals-of-interest (SOI). Because SOI can have low signal-to-noise ratios (S/N), the ability to collect these signals can be compromised by noise generated internally or externally to the site. Currently, sites do not have the means nor the training to measure performance degradation caused by interfering noise levels and system loss. In fact, most NSG sites solely rely upon periodic Signal-to-Noise Enhancement Program (SNEP) team visits to make these performance measurements. Since SNEP visits only occur approximately every three years, the ability to measure SOI collection performance on a more frequent basis must be made available to site personnel.

SNEP team visits, although primarily sponsored by NSG, are not limited to Navy sites. During May 1992, a SNEP team survey was conducted, under Army sponsorship, at a facility located in Augsburg, Germany. Each survey is unique, however, each operates under the premise of improving a site's signal-to-noise ratio. This is done through the tedious work of, first, identifying and locating both internal and external

noise sources, second, identifying system losses, and third, making SOI collection performance measurements using the Performance Evaluation Technique (PET) developed by faculty and students of the Naval Postgraduate School. The resulting information is used to develop actions to mitigate any factor that degrades performance. Finally, all the information obtained during the SNEP survey is consolidated into a final report which is provided to the commanding officer.

Currently, two methods are used to generate PET curves. The first method, or the "manual method", uses construction techniques to establish the desired curves [Ref. 1]. A semi-automated method uses drawing software such as DRAWPERFECT to perform the construction of curves. For the purpose of this report the technique using DRAWPERFECT is considered to be in the same category as the manual method. The second method uses the GRAFTOOL graphics software. Although this system provided benefits over the manual method, it was concluded that further benefits could be obtained through customized software using a high-level language. [Ref. 1] A major portion of this thesis involved the development of PET software using MATLAB. Along with the software development, operating instructions were generated. All three methods were compared and contrasted in four major areas: cost, speed, accuracy and ease-of-use.

II. PERFORMANCE EVALUATION TECHNIQUE (PET)

The Performance Evaluation Technique is a simple, but accurate tool used to provide information concerning the well being of the Radio Frequency Distribution (RFD) system and various receiving systems used by Naval Security Group sites. The PET, through the Signal-to-Noise Enhancement Program, provides a site with a percentage of SOI lost due to noise interference and system deficiencies. Many causes for the loss of SOI have been identified by SNEP visits; the most frequently observed are [Ref. 1]:

- Excessive attenuation in the RFD system.
- High noise floor in the RFD and receiving equipment.
- Site-generated noise and interference within the RFD.
- Saturation of active elements due to the RFD's limited dynamic range.
- Excessive interference and attenuation in cables and connectors due to improper installation and shielding.
- Excessive radio frequency interference (RFI) from internal and external sources.

This chapter describes the process of generating a PET plot. First, the SOI amplitude distribution is described. This is followed by a discussion of each input parameter and the related measurement processes. The construction and interpretation of PET plots using both the manual technique and the semi-automated technique devised by LT Brian Skimmons

[Ref. 1] is also explained. The final section of this chapter summarizes the PET's usefulness to management.

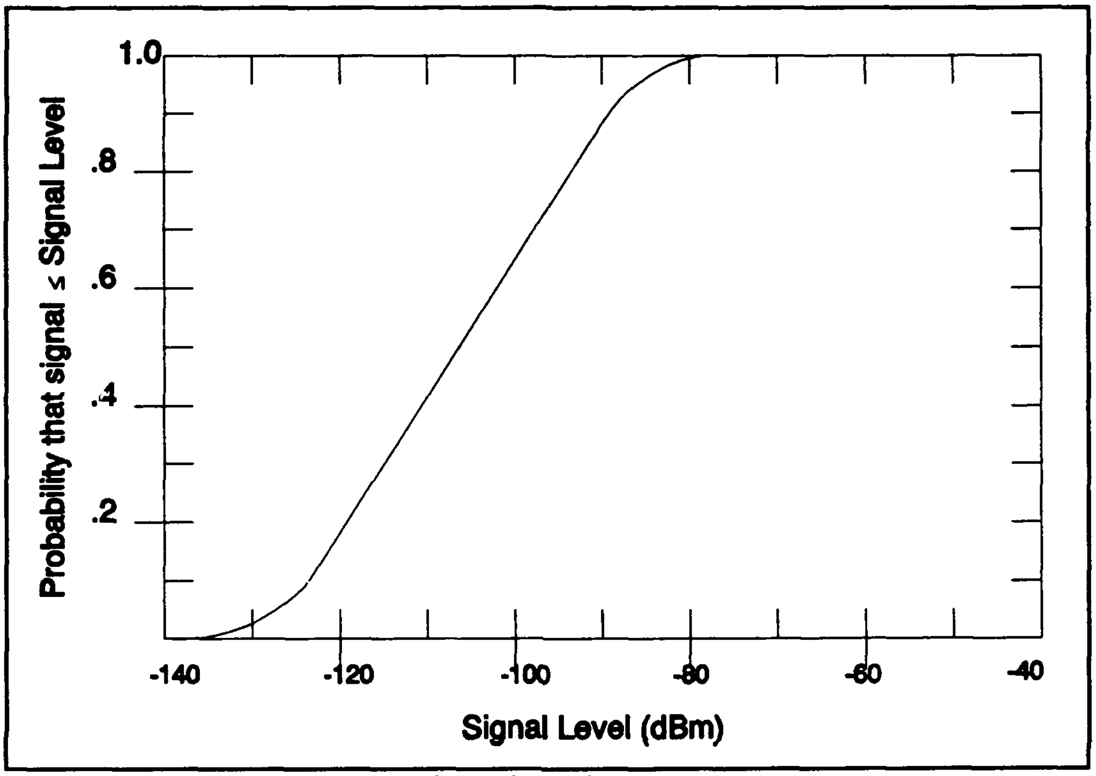

Figure 1 Log-Normal distribution [Ref. 2]

A. SOI AMPLITUDE DISTRIBUTION

Extensive observations and research have established the model for the SOI amplitude distribution. PET accuracy is highly dependent on the validity of the SOI distribution model. Many different studies have been conducted with most suggesting the log-normal distribution shown in Figure 1. Chapter III of the Doctoral Dissertation of LCDR G. Lott, explains why the log-normal distribution is the correct choice for SOI distribution. [Ref. 2]

B. INPUT PARAMETERS

There are 6 different parameters that may be input into the PET curve. These are:

- Noise Floor of a receiving system.
- Maximum received signal strength.
- RFD measured path losses.
- Excess noise floor.
- External and internal Noise levels.
- Measured Signal strength.

Each of these parameters can be obtained through various means; however, the instrumentation configuration used by the SNEP team provides a baseline. This configuration is shown by Figure 2.

1. RECEIVING SYSTEM NOISE FLOOR MEASUREMENT

The system noise floor used by the PET is similar to self-generated, or internal, receiver noise. Every receiving system has a unique noise floor. It is normally provided as a receiver specification; however, it can also be estimated by:

$$N_0 = kT_{eq}B \ , \tag{1}$$

where N_0 is the noise power in watts, T_{eq} represents the equivalent noise temperature in degrees Kelvin, k represents Boltzmann's constant = 1.38×10^{-23} W/K-Hz, and B represents the bandwidth in Hz. [Ref. 3] Using the standard temperature, 290°K, for T_{eq} and a 3-kHz bandwidth, yields a

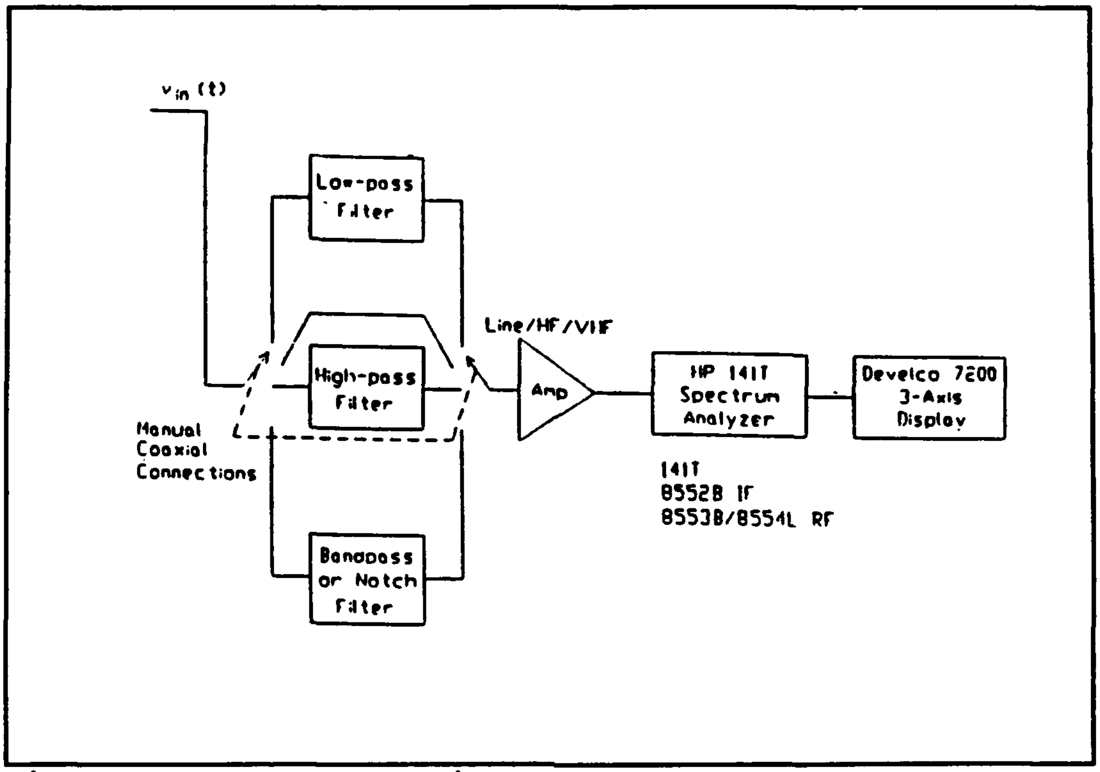

Figure 2 SNEP Team Equipment Setup [Ref. 2]

minimum system noise floor of -139 dBm. Most receivers do not operate with a noise floor this small. If the specified value is not available, a good reference level for a 3-kHz Gaussian-shaped bandwidth is -125 dBm. However, if the specified noise floor is available, it is the preferred value to use. [Ref. 1]

In some cases the specified value may be provided in a different form. One of the most common is the noise figure. Noise figure (F) is the ratio of total output noise to input noise, when the input source is at the standard temperature.

$$F=\frac{k(290^{\circ})B+N_{internal}}{k(290^{\circ})B} \tag{2}$$

The previous equation shows that the output noise is the input noise plus whatever device noise, $N_{internal}$, is added. From the noise figure, one can determine the equivalent temperature, T_{eq}, from:

$$T_{eq}=(F-1)290^{\circ} \quad . \tag{3}$$

Using this temperature and equation (1), the system (receiver) noise floor can be obtained. [Ref. 3]

In order to use the system noise floor as a parameter for PET analysis, it must be in dBm. This parameter is mandatory for all PET analysis. It provides the lower limit for the SOI amplitude distribution curve. No signals can be intercepted below the system's noise floor. Actual use of the parameter is discussed in section C of this chapter. [Ref. 1]

2. MAXIMUM RECEIVED SIGNAL STRENGTH

The maximum signal strength parameter is the strongest signal strength predicted within the HF band (2-32 MHz) and the strongest SOI signal strength, as a function of time and bearing. This parameter is very dependent on ionospheric conditions and path geometry.

Currently the SNEP team utilizes HF prediction software, such as Advanced PROPHET developed by the Naval Ocean Systems Center [Ref. 4]. Additional software programs such as SORFMS and IONCAP have also been used. Prior to visiting a site, SNEP team members generate a list of known transmitting sites which are likely to provide the maximum

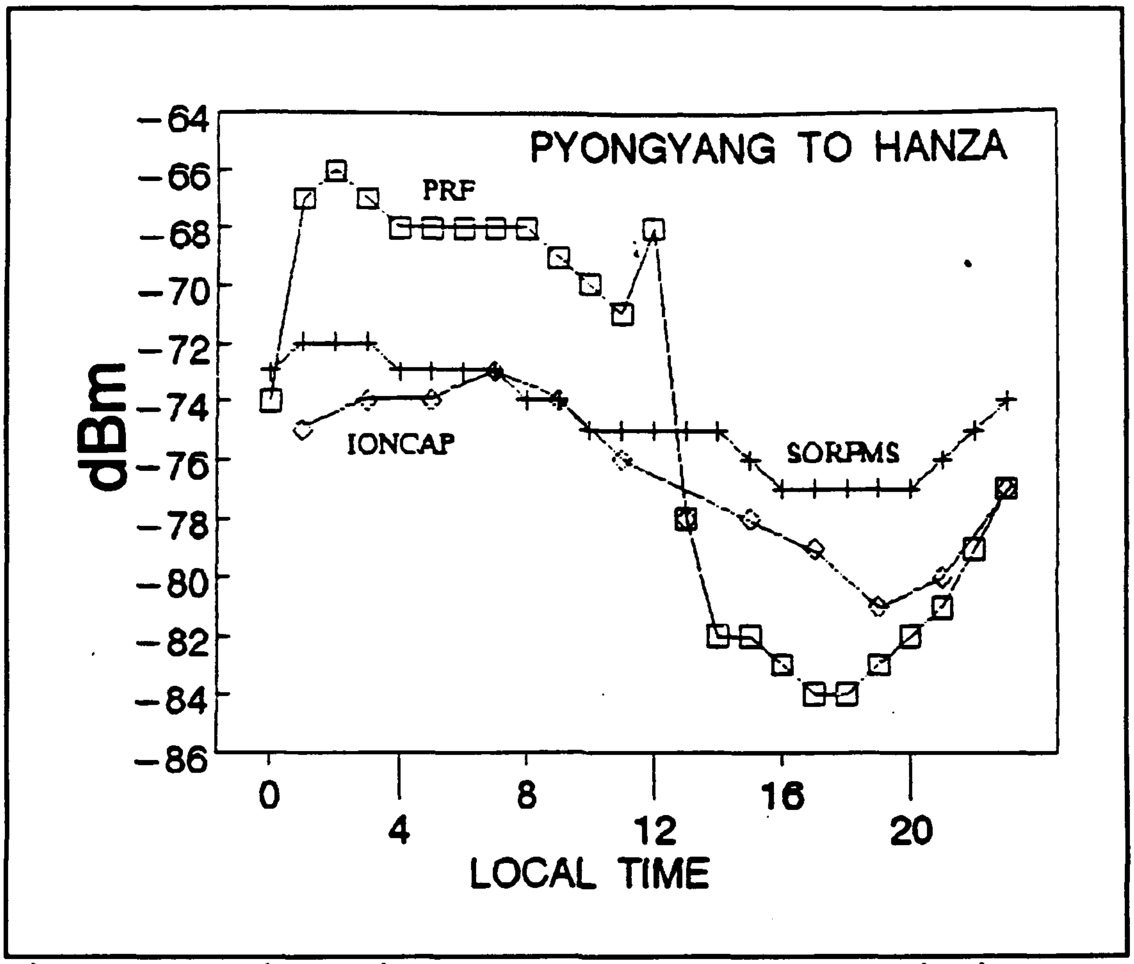

Figure 3 Maximum Signal Strength Software Predictions [Ref. 5].

signal strength received by the site being surveyed. Normally this list of transmitters is made up of large, high-powered international shortwave broadcasting sites (such as Radio Moscow) and selected sources of SOI. Required data is entered and the software provides predicted signal amplitudes for the HF spectrum over 24 hours of the selected day. Figure 3 displays the differences in software predictions as well as maximum signal strength variation with time and frequency. [Ref. 1]

Alternatively, the maximum signal strength can be measured directly by means of the equipment described in Figure 2. An operator can cycle through the 2-32 MHz band and record the largest signal strength displayed by the spectrum analyzer at the selected bearing and time. This method can be very time consuming, especially when measuremnts are required at hourly intervals over a period of several days to obtain good statistics. The measurement points for maximum signal strength, as well as other important parameters used by the PET are shown in Figure 4.

Maximum SOI signal strength in dBm sets the upper limit of the SOI distribution curve and establishes a basis for all amplitudes generated by the PET. Use in PET analysis is discussed in the following section.

3. RFD LOSSES

RFD loss is a site-specific parameter that can only be obtained through measurement. The RFD includes all components between the antenna termination plates and the input to a specified receiving system. [Ref. 5] In Figure 4, the RFD loss is the combination of the primary multicoupler (PMC) loss, the cable loss, and Enlarger loss. Enlarger provides multiple users with automated switched access to the antenna beams. It is used by the majority of NSG sites. [Ref. 1]

RFD losses are determined by injecting a test signal of known power and frequency and measuring the loss in signal

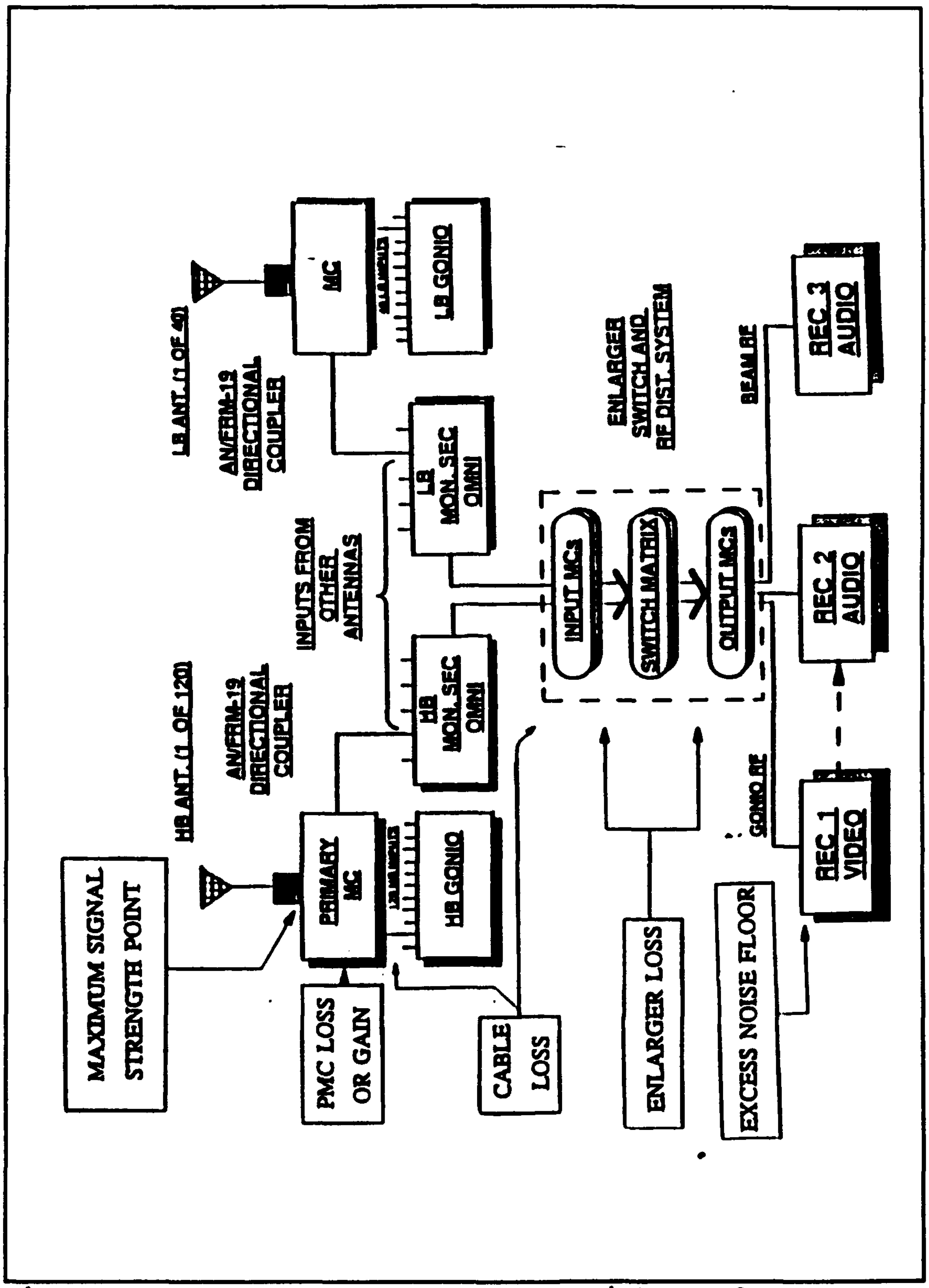

Figure 4 PET Parameter Measurement Points [Ref. 1]

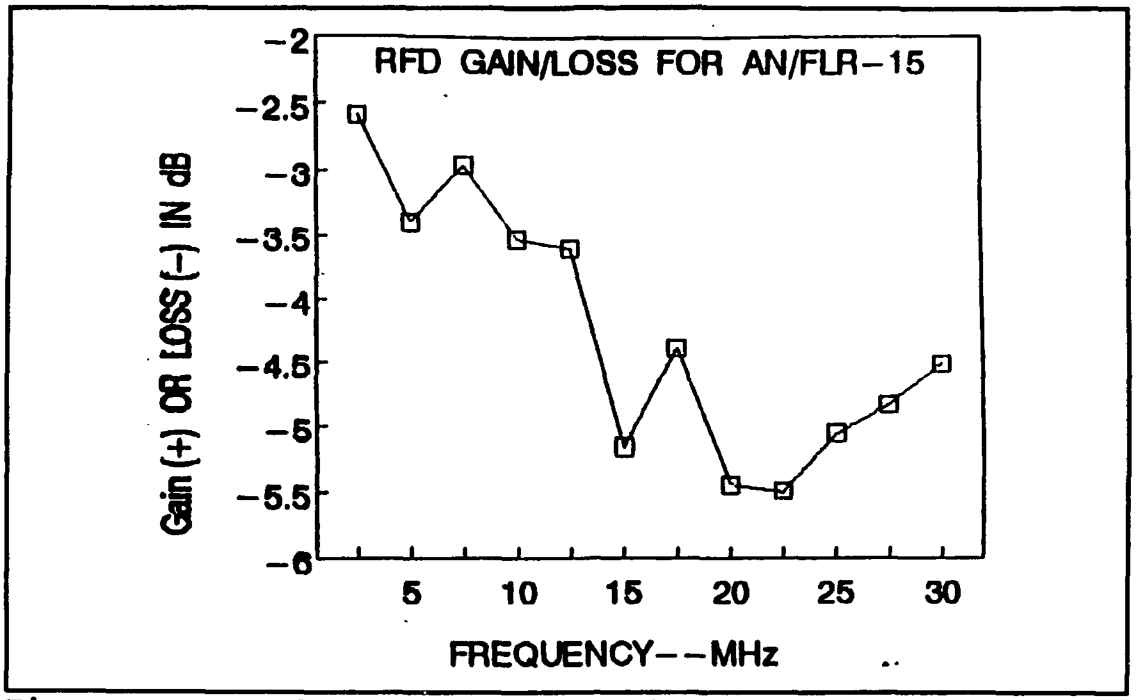

Figure 5 Example of RFD Loss [Ref. 5]

level at one of the measurement points in Figure 4. Because RFD loss often varies with frequency, the entire HF band is examined. Figure 5 shows an example of the variation in loss with frequency. Most NSG RFD systems divide the full azimuth into 12° degree monitor beams. These RFD systems, therefore, have 30 unique paths for received energy to travel. Currently, the SNEP team measures losses for a few beams which are assumed to be representative of the other paths. In many cases the bearing-dependent loss varies by less than a few dB; however, there are sites where the measured RFD loss varies significantly with bearing.

For PET use, RFD loss is measured in dB. In some situations, RFD gain may occur; however, the gain does not improve performance. RFD gain can abnormally affect

performance by increasing dynamic range requirements. A net RFD gain near 0 dB is the most desirable condition. [Ref. 5]

4. EXCESS NOISE FLOOR

Excess noise floor is defined as an increase in noise floor over the specified system noise floor and can be produced by components within the RFD system. A primary source of excess noise floor is the active elements of the ENLARGER system mentioned previously. Figure 6 is an example of excess noise floor measurements. [Ref. 1]

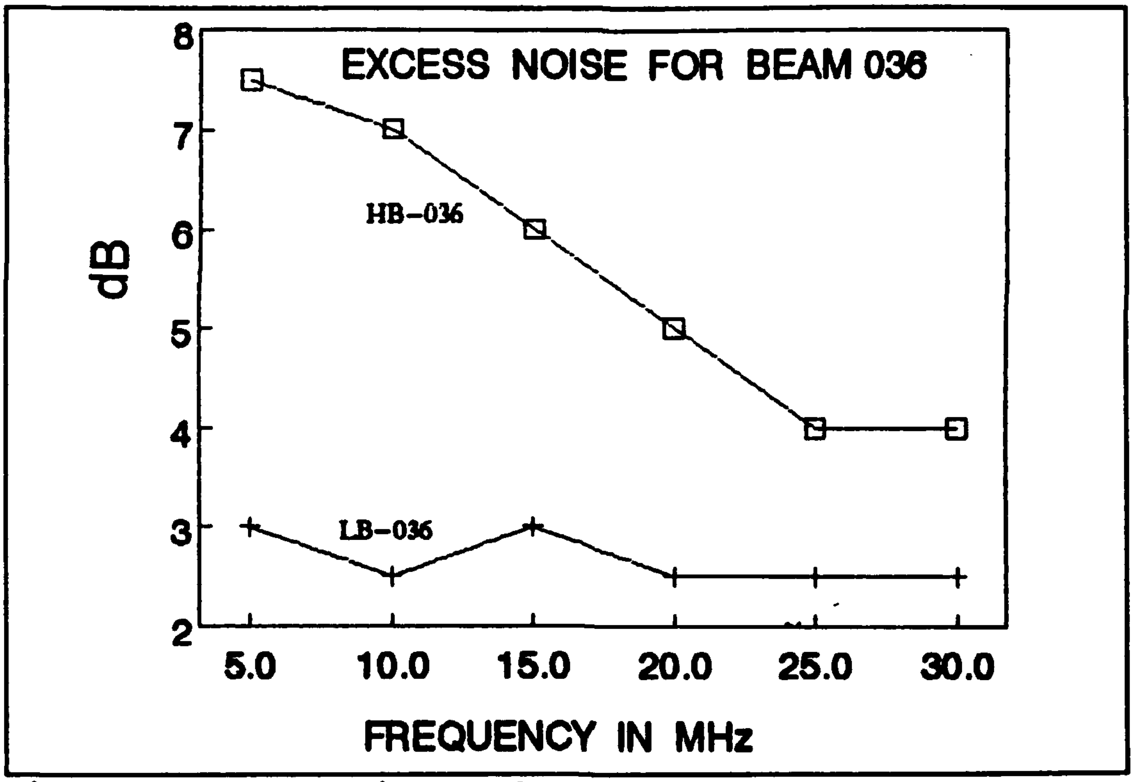

Figure 6 Excess Noise [Ref. 5]

The excess noise floor parameter can only be obtained through measurements where all inputs to the selected RFD path are terminated, thereby, eliminating external noise sources

which would enter through the antenna. The excess noise floor is obtained by using a spectrum analyzer to identify a frequency range where minimal interfering signals are present. The lowest power level observed is the level of the excess noise floor. The difference between this and the system specified noise floor is the excess noise parameter. Excess noise typically varies with frequency. Again, since there are 30 different paths associated with the 12° monitor beams, excess noise can vary with bearing. Currently, the SNEP team measures noise for a few bearings and uses an average value to represent the remainder. [Ref. 1]

Excess noise is expressed in dB units and is used as described in section C of this chapter.

5. MAN-MADE NOISE SOURCE LEVELS

Noise adversely affects a site's ability to receive signals. Basically, there are two different types of noise observed during SNEP team surveys. Depending on their origin, they are categorized as either internal or external sources of man-made noise. Although the excess noise floor is also a form of man-made noise, it is considered separately. The excess noise floor is caused by active devices in the RFD with high noise figures.

Internal man-made noise is caused by improperly operating devices within the RFD and by other noise sources that are coupled into the RFD. Typical internal sources are

those sources within a site, such as computers, uninterruptible power supplies (UPS), frequency converters, telephone switches, and other equipment capable of generating radio interference. The noise generated by these devices is conducted along conductors within the site and is in turn, conducted or inductively coupled into the RFD system and subsequently a receiver. Internal noise can travel along any conductive material. Some examples include power conductors, cable shields, grounds, air-conditioning ducts, building structural material, and other conductors inside the site. Because of these complex conducting paths, internal noise sources are often very difficult to identify and isolate. [Ref. 1]

The SNEP team attempts to isolate and eliminate, or mitigate, any internal noise sources discovered. However, if mitigation is impossible, the noise power becomes an input parameter used by the PET. Internal noise sources vary with frequency, time and even bearing. Examples of internal noise are shown in Figures 7 and 8. Figure 7 shows an internal noise source found on a bearing of 036 at frequencies between 2 and 4 MHz and Figure 8 shows a source bearing 348 at 3.49 MHz. Both figures demonstrate that internal noise sources vary with frequency and bearing. [Ref. 5]

External noise sources are man-made noise sources external to the site. Examples of external noise include: power-line noise, ignition noise, or out-of-band emanations

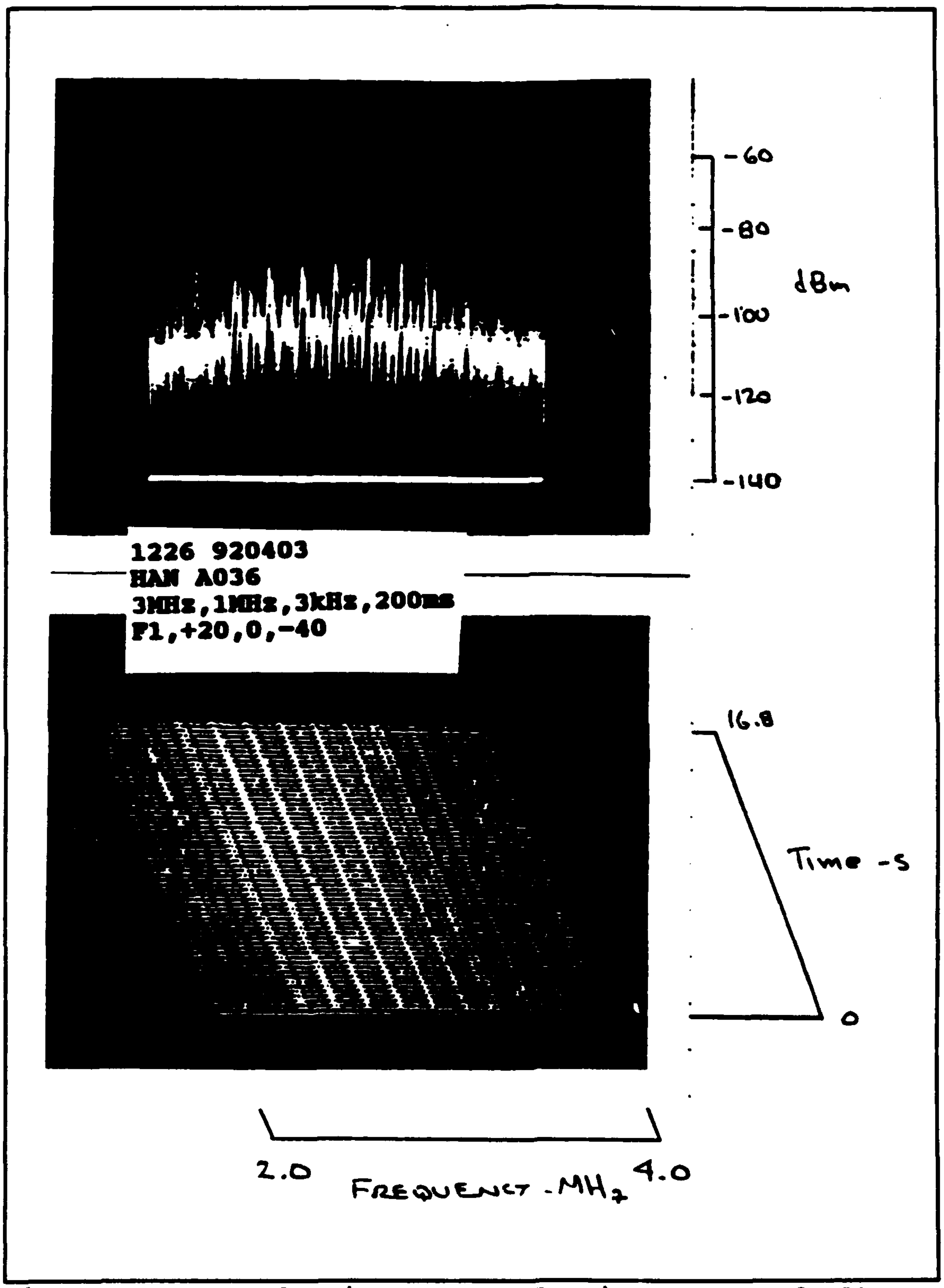

Figure 7 Internal Noise Source (Bearing 036) [Ref. 5].

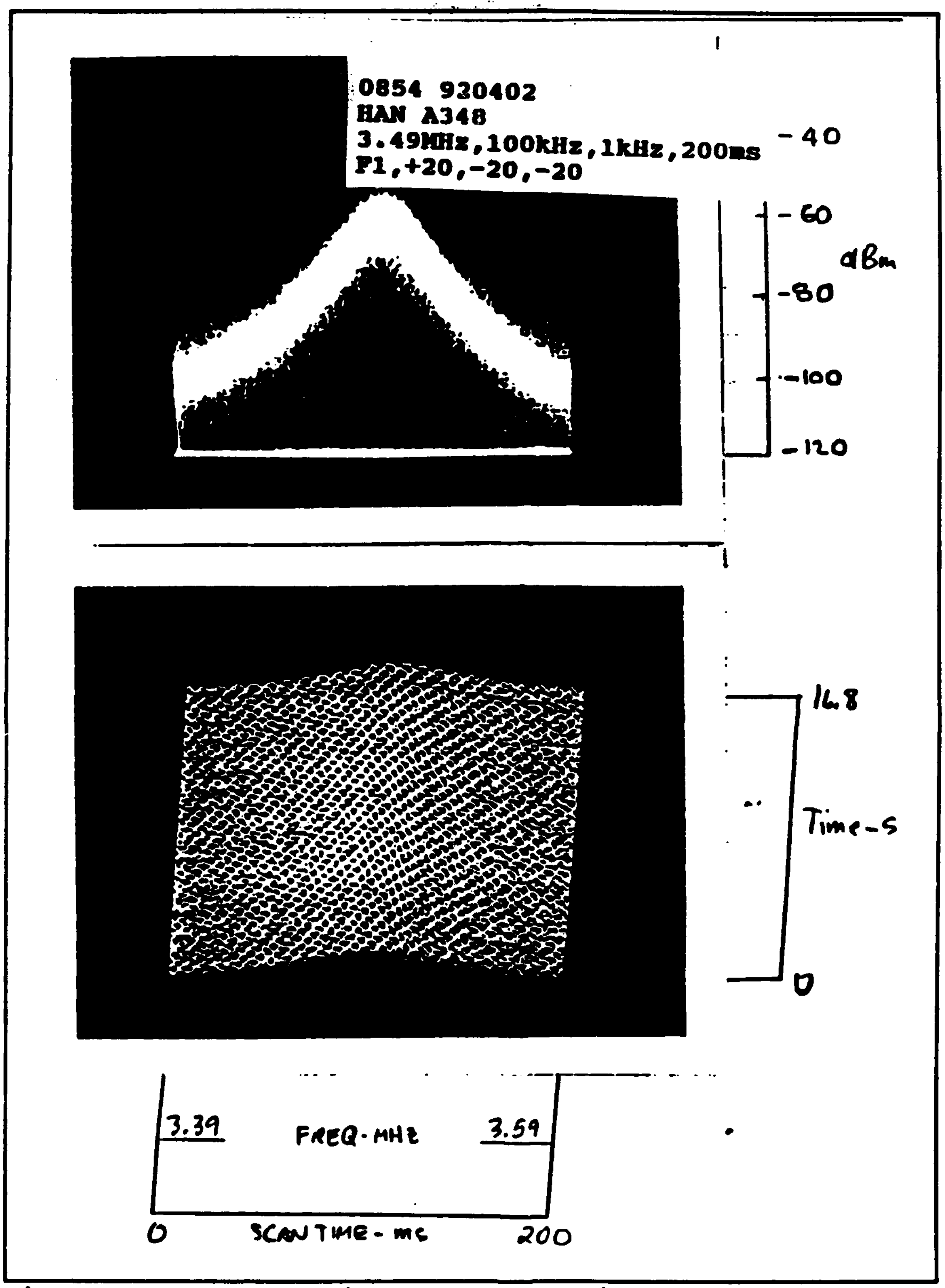

Figure 8 Internal Noise Source (Bearing 348) [Ref. 5].

from industrial medical and scientific (ISM) equipment. Power-line noise is by far the most prevalent. Examples of both power-line noise and ignition noise are shown in Figures 9 and 10.

As with internal noise, external noise also varies with time, frequency, and bearing. Some sources of external noise can be very difficult to localize because the noise may have traveled over one or two ionospheric hops. The large geographical areas involved may complicate the localization process and prevent rapid isolation of such sources. Other sources of external noise, such as sources associated with power lines, are usually within line-of-sight from the uppermost part of a receiving antenna. These sources can be located and eliminated.

For use in the PET curves, noise source level, whether internal or external, must be converted to dBm. It will be shown through PET analysis demonstrated in section C, that noise power collected by the antenna's elements is a major concern to site managers.

6. KNOWN SIGNAL STRENGTH

Signal strength can be thought of as the received power of some transmitting source. PROPHET or some other HF statistical prediction software can be used to estimate the expected value of power received at a specified site. Site personnel can use the PET curve to determine, on the average,

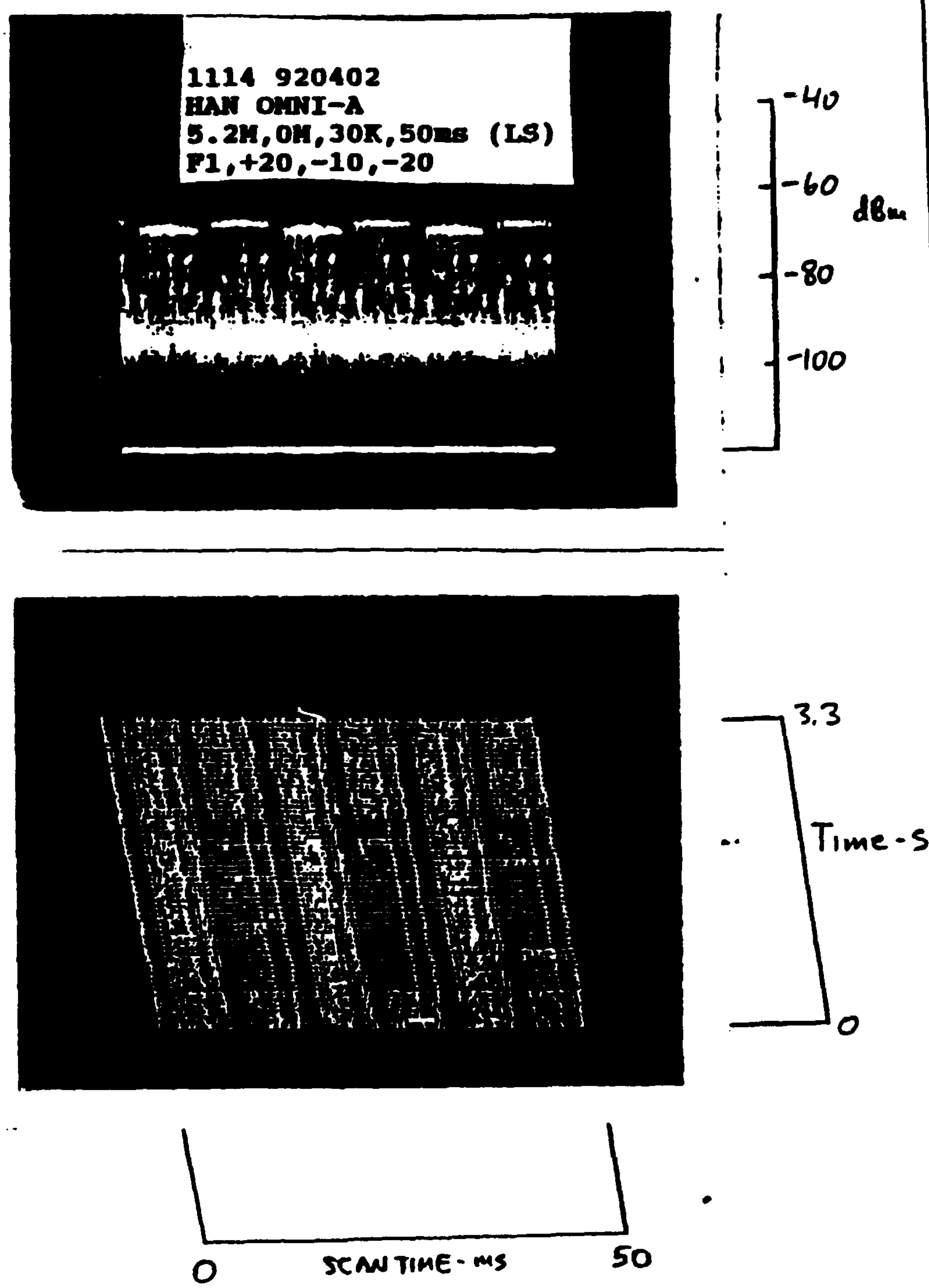

Figure 9 Example of Power Line Noise [Ref. 5].

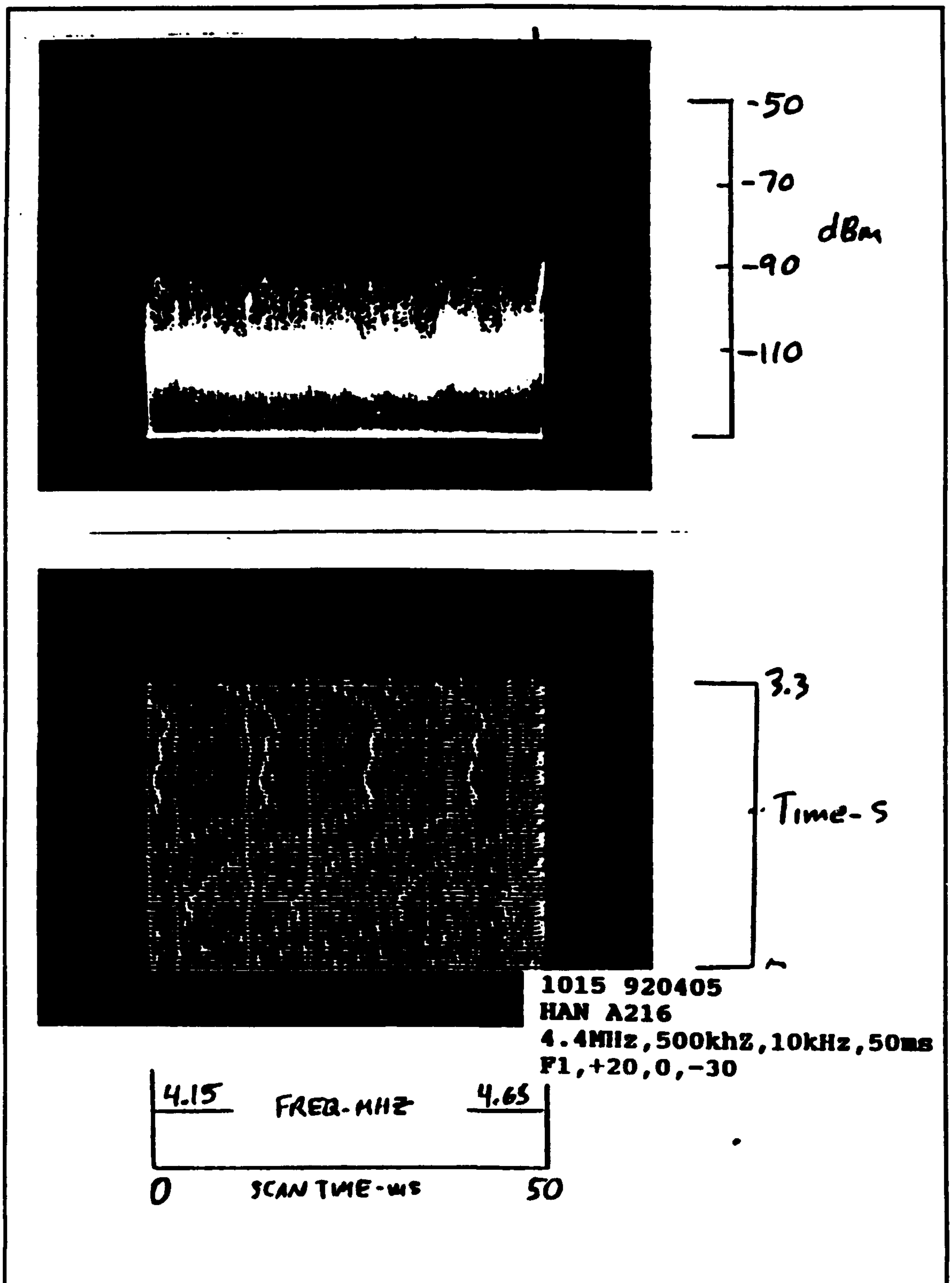

Figure 10 Ignition Noise [Ref. 5].

whether the known signal could be intercepted. Such a signal, for example, might originate from a submarine using a 1000 watt transmitter and a monopole antenna operating on 5 MHz. PROPHET can be used to estimate the received signal strength as a function of time. This signal strength value would be input into the PET and a determination made as to whether, on the average, the signal could be intercepted. This is demonstrated in the following section.

C. MANUAL PET CONSTRUCTION

Manual PET curves can be constructed using pencil and graph paper or a computer drawing program, such as DRAW-PERFECT. This section describes how to produce PET curves manually.

1. STEP 1: SOI AMPLITUDE DISTRIBUTION LINE

As shown in section A, the SOI amplitude distribution is log-normal. In order to simplify manual production of PET curves, a linear approximation to the log-normal distribution is used. Figure 11 demonstrates the close agreement between the straight line and the log-normal curve, except at the tail ends of the distribution which are of little interest. [Ref. 1]

The first step in the manual construction process is establishing the SOI amplitude distribution line, shown in Figure 12. The x-axis is labeled Signal Strength (dBm). Make the far left point of the x-axis smaller than the system noise

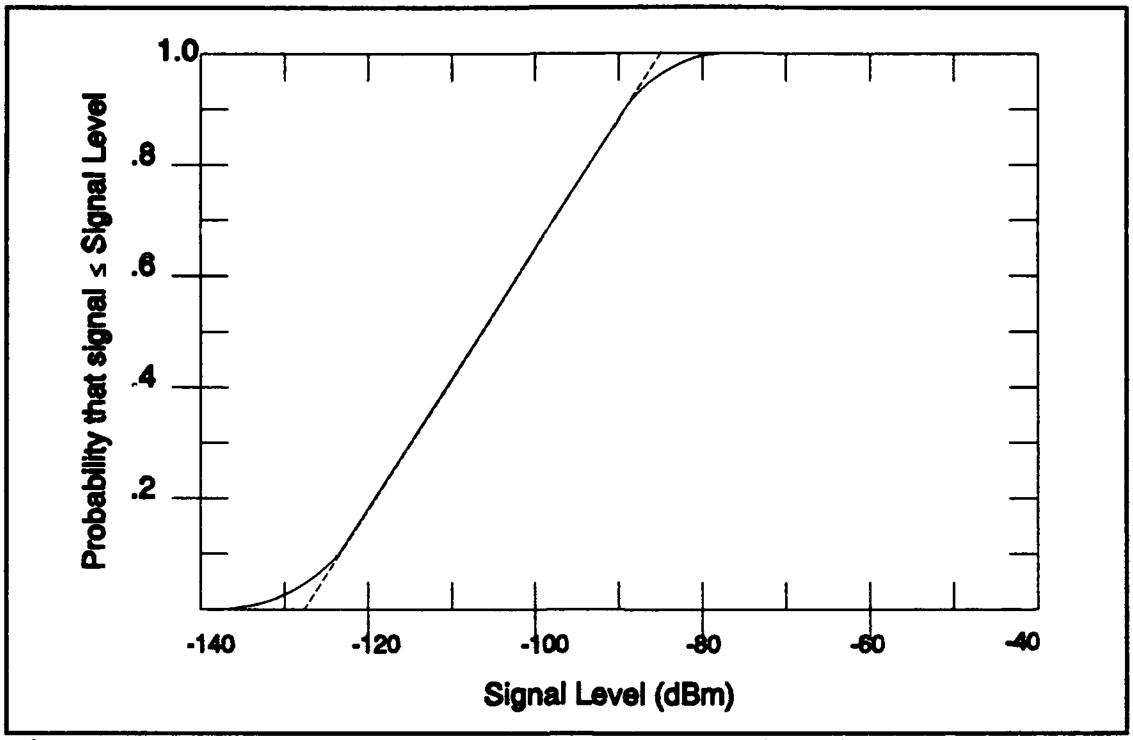

Figure 11 Log-normal vs. Linear approximation [Ref. 1].

floor in dBm. Make the far right point of the x-axis greater-than-or-equal to the maximum signal strength, also in dBm. The y-axis is labeled Percent SOI Available and ranges from 0 to 100%. Draw a straight line from the (maximum signal strength, 100%) point to the (system noise floor, 0%) point. This line is the SOI amplitude distribution.

The interpretation of the SOI amplitude distribution line is quite simple. Enter a specific power level on the horizontal x-axis. Go vertically until the SOI amplitude distribution line is intercepted. Then proceed horizontally and read off the percentage. This is the percent of the available SOI with received signal levels between the system noise floor and the entered signal strength. It is evident

that 100% of all SOI available fall below the maximum signal strength and no SOI is available below the system noise floor.

Figure 12 also provides an example based on a system noise floor of -125 dBm, a maximum signal strength of -80 dBm and an entered signal strength of -97.5 dBm. Using the linear approximation, 64% of the SOI available fall between -95 dBm and -125 dBm and 36% fall between -95 dBm and -80 dBm.

2. STEP 2: +12 dB DISTRIBUTION LINE

The +12 dB detection offset distribution line is established to represent the Signal-to-Noise value for automated detection and processing of SOI required by most conventional digital processing techniques. [Ref. 1]

Refer to Figure 13 for the construction of the +12 dB distribution line. To enter the line on the PET plot, draw a line parallel to the SOI amplitude distribution line shifted 12 dB to the left. In the example described in Figure 12, (system noise floor of -125 dBm and maximum signal strength of -80 dBm), the points (-137 dBm, 0%) and (-92 dBm, 100%) are connected.

The interpretation of the +12 dB distribution is exactly the same as that described for the SOI amplitude distribution line. Rather than providing percent SOI available, the +12 dB line provides SOI available that are +12 dB over the system noise floor. If a user prefers a different detection offset, it is easily accomodated in PET. [Ref. 1]

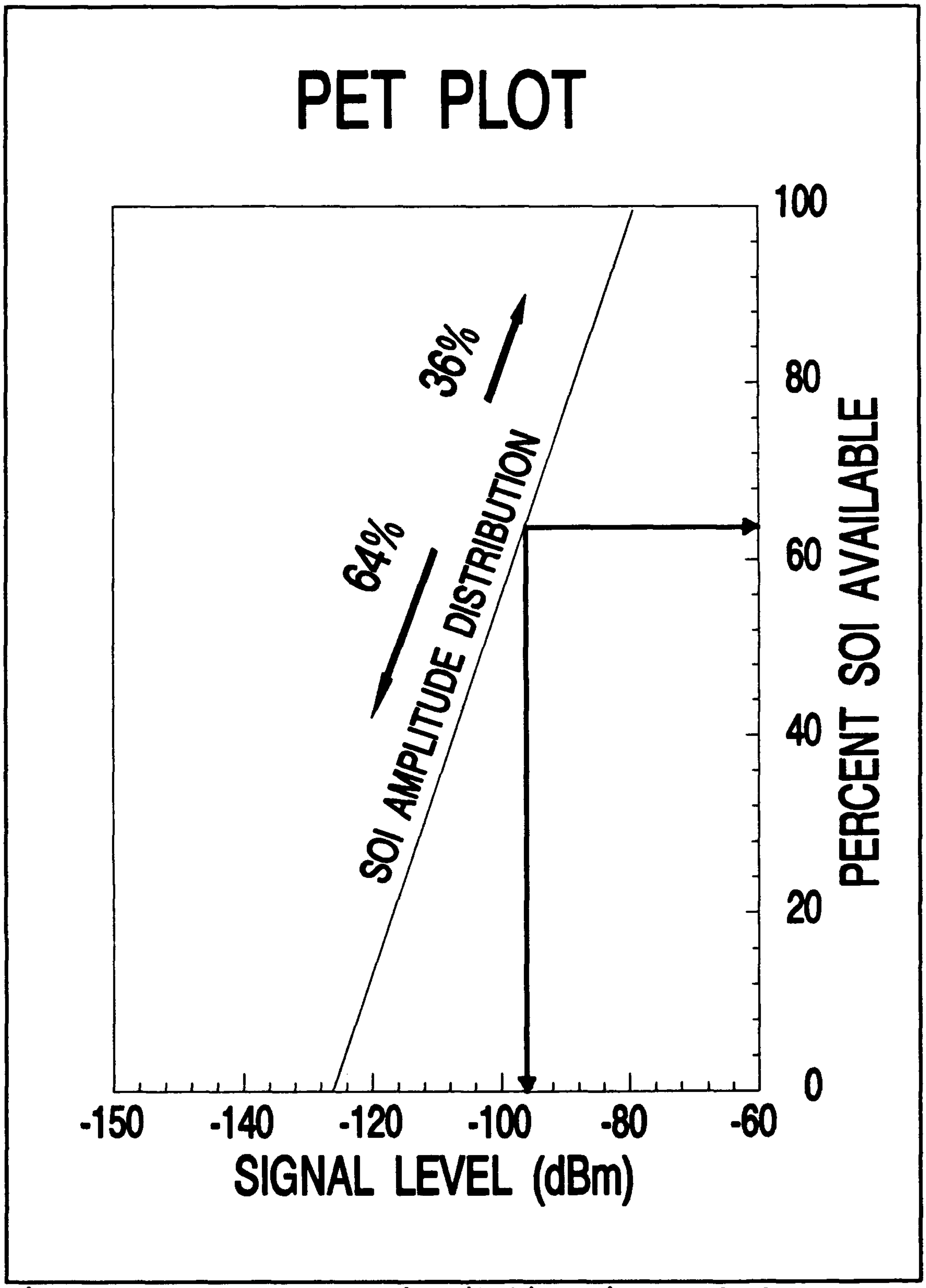

Figure 12 STEP 1: SOI Distribution Line [Ref. 1].

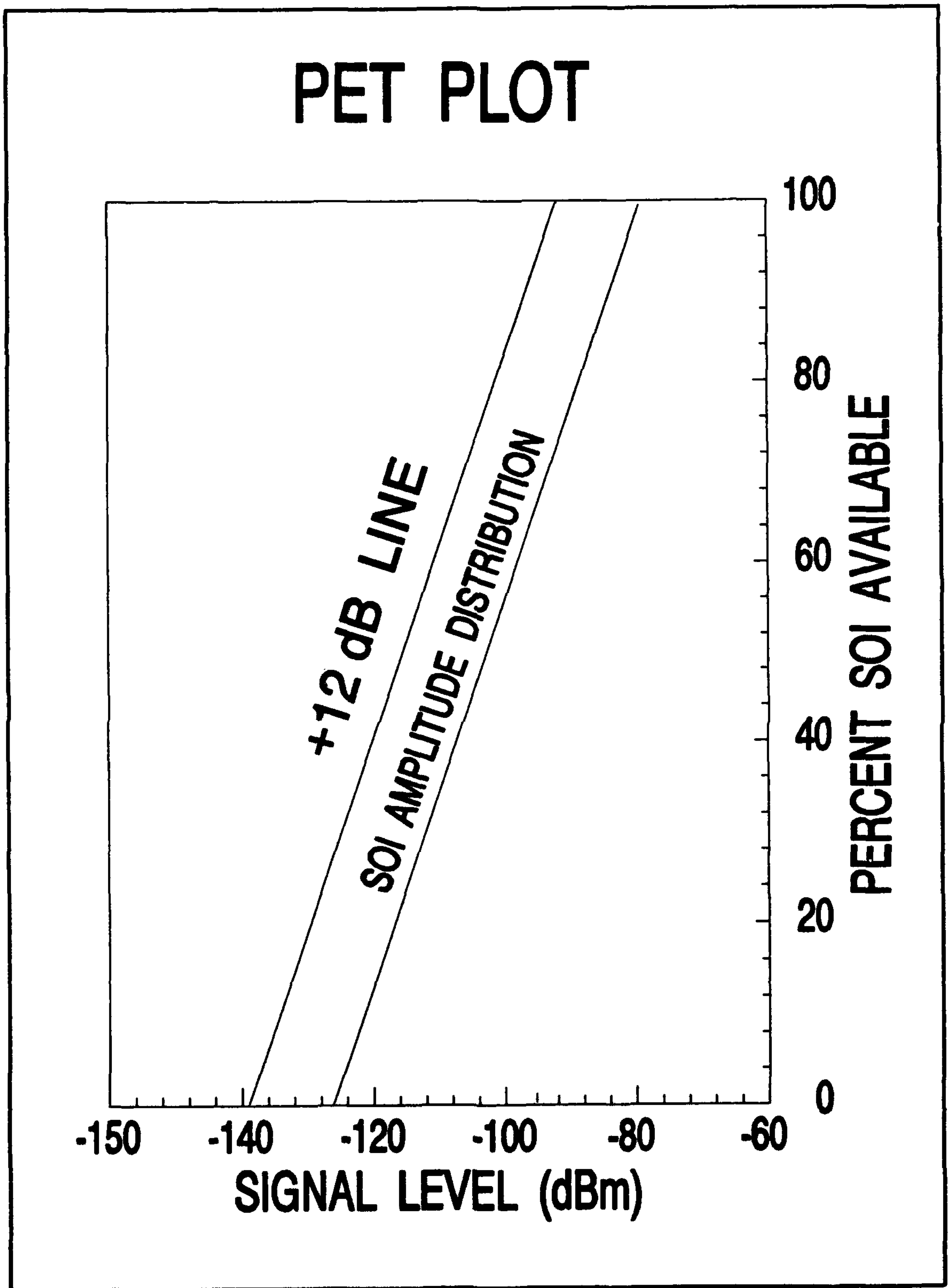

Figure 13 +12 dB Distribution Line [Ref. 1].

3. STEP 3: SITE PERFORMANCE LINE

To establish the site performance line, construct a new scale on the y-axis, *representing the Percent SOI Lost* (Refer to Figure 14). Construct a vertical line from the system noise floor value (-125 dBm) to the intersection with the +12 dB distribution line. Next, construct a horizontal line from the intersection point to the new y-axis scale. The y-axis value represents the 0% SOI lost level. The 100% SOI lost level is located at the same level as the 100% available SOI level.

Refer to Figure 15 to establish the Site Performance line which is parallel to the SOI distribution line and shifted to the left by the measured signal loss in the RFD in dB. Like the +12 dB detection offset distribution line, the Site Performance line is shifted only to the left. The Site Performance Line represents the optimum performance the site can maintain with the measured RFD loss. If the system has a measured RFD gain/loss of 0 dB, the +12 dB detection offset distribution line becomes the Site Performance line. This is true because, as discussed earlier, the RFD loss parameter is set to zero and therefore, no shifting to the left occurs. If an RFD gain greater than zero is entered into the PET, the Site Performance line would have to be shifted to the right. The example shown in Figure 15 has zero RFD loss or gain.

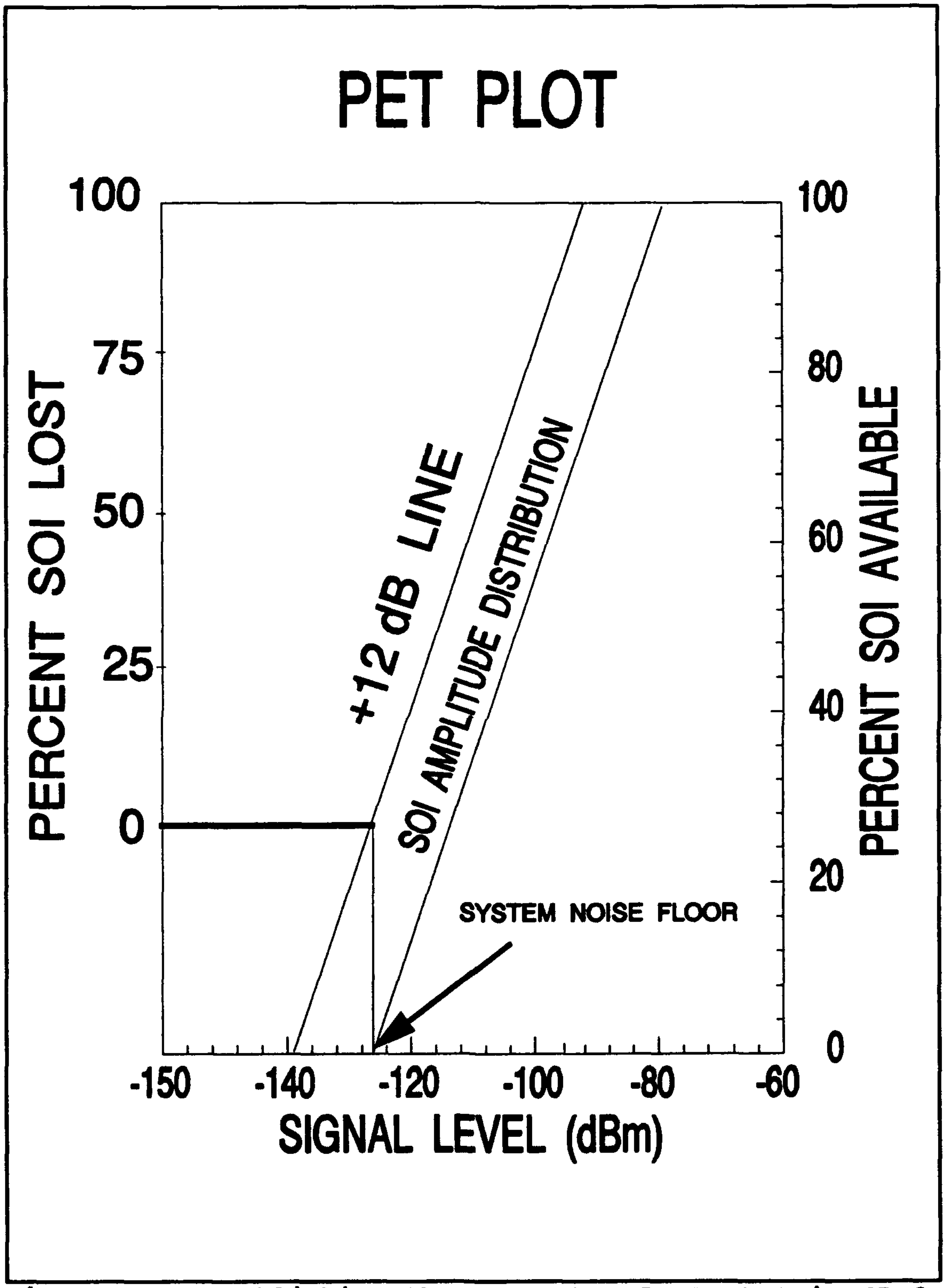

Figure 14 Establishing the Percent SOIs Lost Axis [Ref. 1].

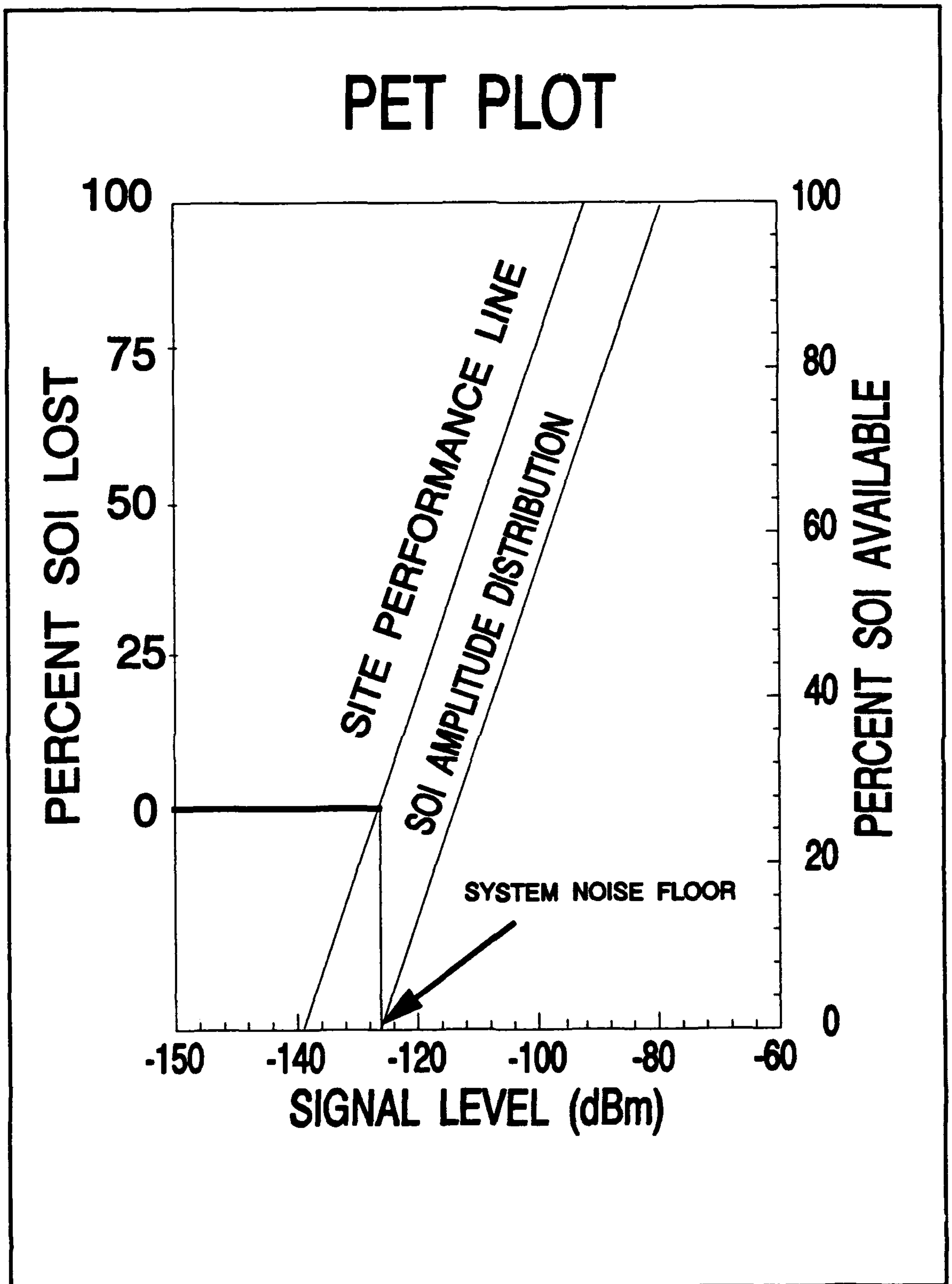

Figure 15 Site Performance Line [Ref. 1].

4. STEP 4: SIGNALS LOST DUE TO RFD LOSSES

Once the Site Performance line has been constructed, the PET plot becomes capable of producing the useful results for which it was designed. To determine the percent of signals lost due to RFD losses, first, shift the Site Performance line to the left by an amount equal to RFD loss in dB. Then, draw a vertical line from the system noise floor to the Site Performance line. At this intersection, draw a horizontal line to the Percent SOI Lost axis. This value is the percent of available SOI lost due to RFD losses. In the example presented by Figure 16, approximately 30% of the available SOI were lost due to a measured RFD loss of 10 dB.

5. STEP 5: SIGNALS LOST DUE TO EXCESS NOISE FLOOR

Available SOI can also be lost because of an increase in system noise floor. To determine a loss of signals due to excess noise, enter the PET plot at the system noise floor value. Move horizontally to the right by the amount of excess noise in dB. Draw a vertical line to the Site Performance line. At this intersection draw a horizontal line to the percent SOI lost axis. This value is the approximate percentage of SOI lost due to excess noise. In the example displayed by Figure 17, over 50% of the available SOI were lost due to a combination of excess noise floor of 7 dB and RFD loss of -10 dB.

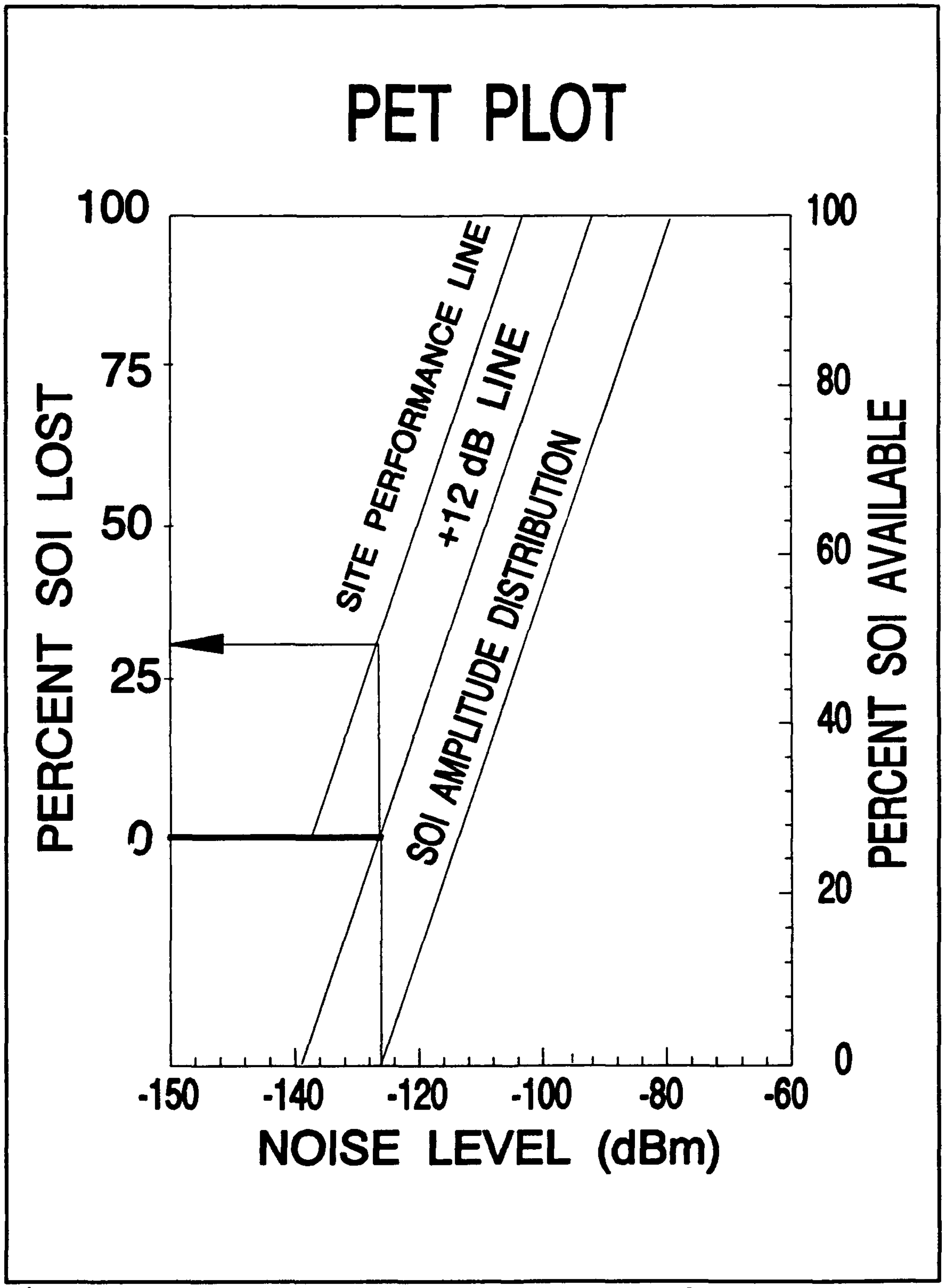

Figure 16 SOIs Lost due to RFD Losses [Ref. 1].

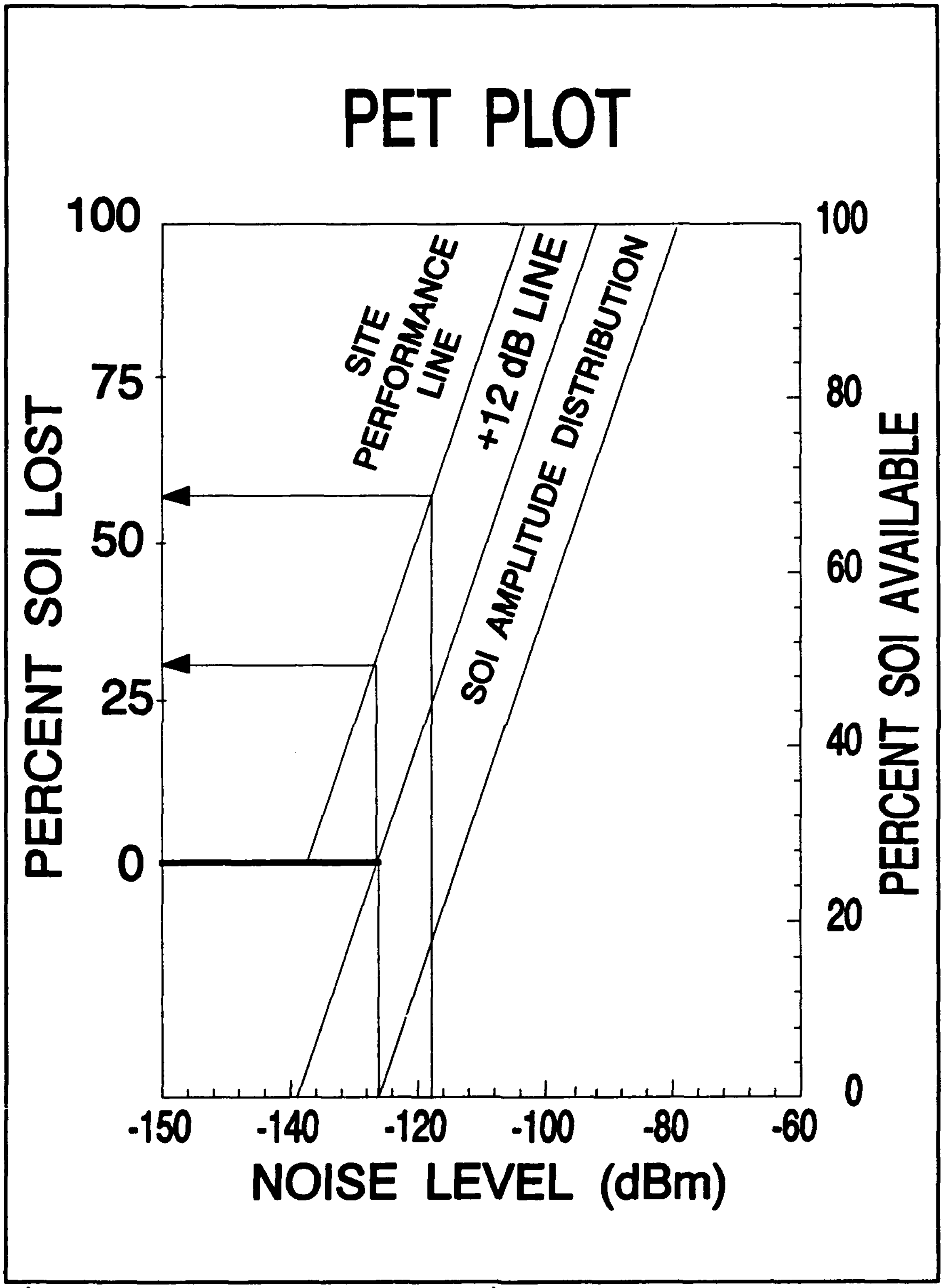

Figure 17 SOIs Lost due to Noise Sources

6. STEP 6: SIGNALS LOST DUE TO INTERNAL/EXTERNAL MAN-MADE NOISE

The PET plot can also be used to identify the percentage of SOI lost due to man-made internal and external noise. Both internal and external noise are treated in the same manner. Enter the PET plot at the value of noise source level in dBm. Draw a vertical line to the Site Performance line. At this intersection, draw a horizontal line to the percent SOI lost axis. This value provides the percentage of SOI lost due to the entered noise level. In Figure 18, over 75% of all SOI available were lost due to a man-made noise level of -108 dBm (measured within the receiver's 3 kHz Gaussian bandwidth).

7. KNOWN SIGNAL STRENGTH PET ANALYSIS

The PET plot can be used to determine whether or not signal reception is likely, on the average. In order to make an accurate estimation of SOI reception, the bearing to the transmitter, the transmitted or received power, and the frequency band for the SOI are required. Using the PET plot for the bearing and frequency of the SOI, draw a vertical line at the observed signal strength of the SOI. If this line, is the rightmost vertical line, then the signal can be intercepted. If it is not, then, on the average, the system is not capable of receiving a signal on that frequency and bearing at that power level.

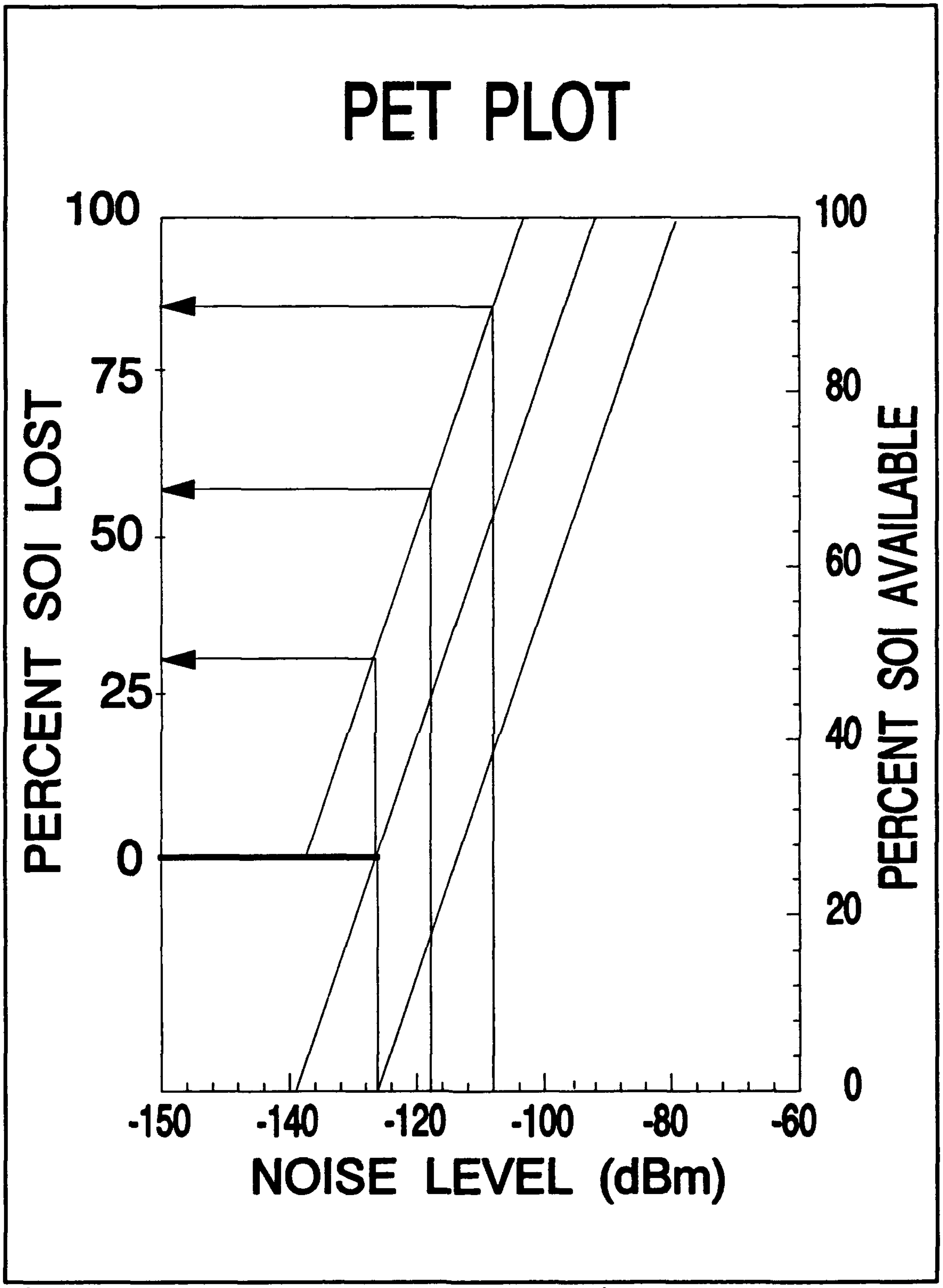

Figure 18 SOIs Lost due to Excess Noise [Ref. 1].

D. GRAFTOOL SEMI-AUTOMATED TECHNIQUE [Ref. 1]

The GRAFTOOL method described by Skimmons as an automated method will be referred to as a semi-automated method in this thesis. In order to use this method, a user must be highly proficient with the GRAFTOOL software. PET plots are constructed using four different data files and various GRAFTOOL commands. The four required data files contain data for plotting the SOI distribution, the +12 dB distribution, and the Site Performance line. Since PET plots differ with frequency and time, each different frequency and time period have corresponding sets (of four) data files. The sample time period used by Skimmons' technique is every four hours and the frequency range selected is 2-30 MHz. Since there are 6 different time periods per frequency and 29 different frequencies, 6*29*4 = 696 different data files are required to make a complete PET analysis of a site. This does not take into consideration the fact that PET analysis can also differ with bearing. [Ref. 1]

Rather than explain in detail each of the computational steps for generating a single PET plot, a general overview is provided covering how these curves are established using the GRAFTOOL method. If a more in-depth discussion is required, see Reference 1.

The first step in the GRAFTOOL method is the setting up of the axes. The Percent Available SOI axis is setup first, then

the Percent SOI Lost axis. The plot should look like Figure 19. [Ref. 1]

The next step involves the creation of the four data files and the plotting of each of the three lines. First, the 0% (system noise floor) and 100% (maximum signal strength) points are entered into data file 0. Second, a GRAFTOOL linear regression command is used to plot the SOI distribution line. The data is stored in data file 1. Third, the +12 dB line is plotted by using a formula command that allows for the subtraction of 12 from each of the points found in data file 1. The new points are stored in data file 2 and the +12 dB line is plotted. Finally, the Site Performance line is generated. This is done by changing the usable axis from percent available to percent lost. Data file 3 is created and contains the two points at 0% (system noise floor) and 100% (maximum value of +12 dB line) using the percent lost axis. Since the Site Performance line is first plotted without loss it is considered the same as the +12 dB line. The linear regression command is once again used with the points placed in data file 4. The resulting plot is shown by Figure 20. In this case, the Site Performance line coincides with the +12 dB line.

RFD loss can be input into the GRAFTOOL method by using the same formula command used to create the +12 dB distribution line. The amount of RFD loss is subtracted from each of the points held in data file 4, and the shifted Site

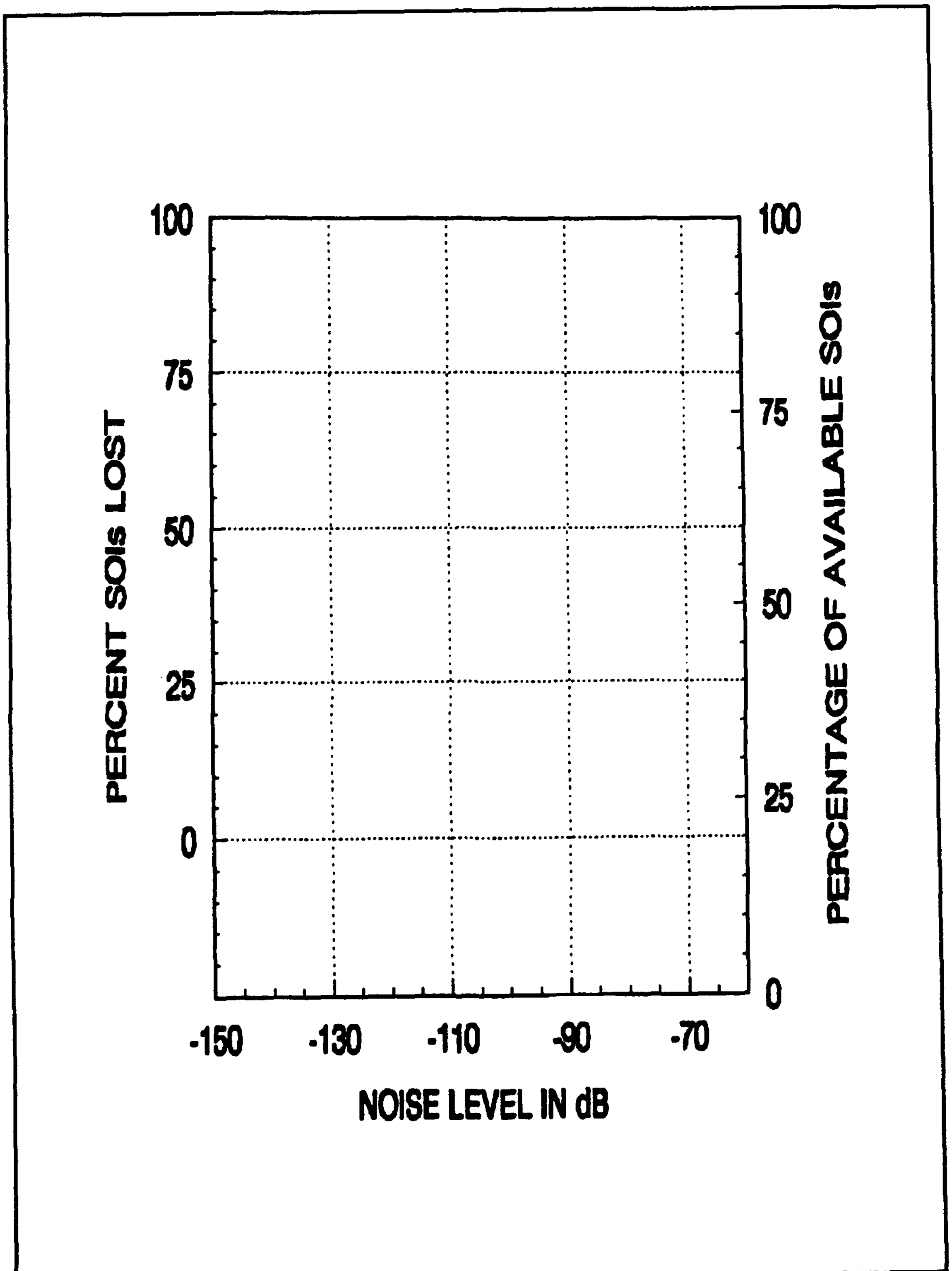

Figure 19 GRAFTOOL Axis Setup [Ref. 1]

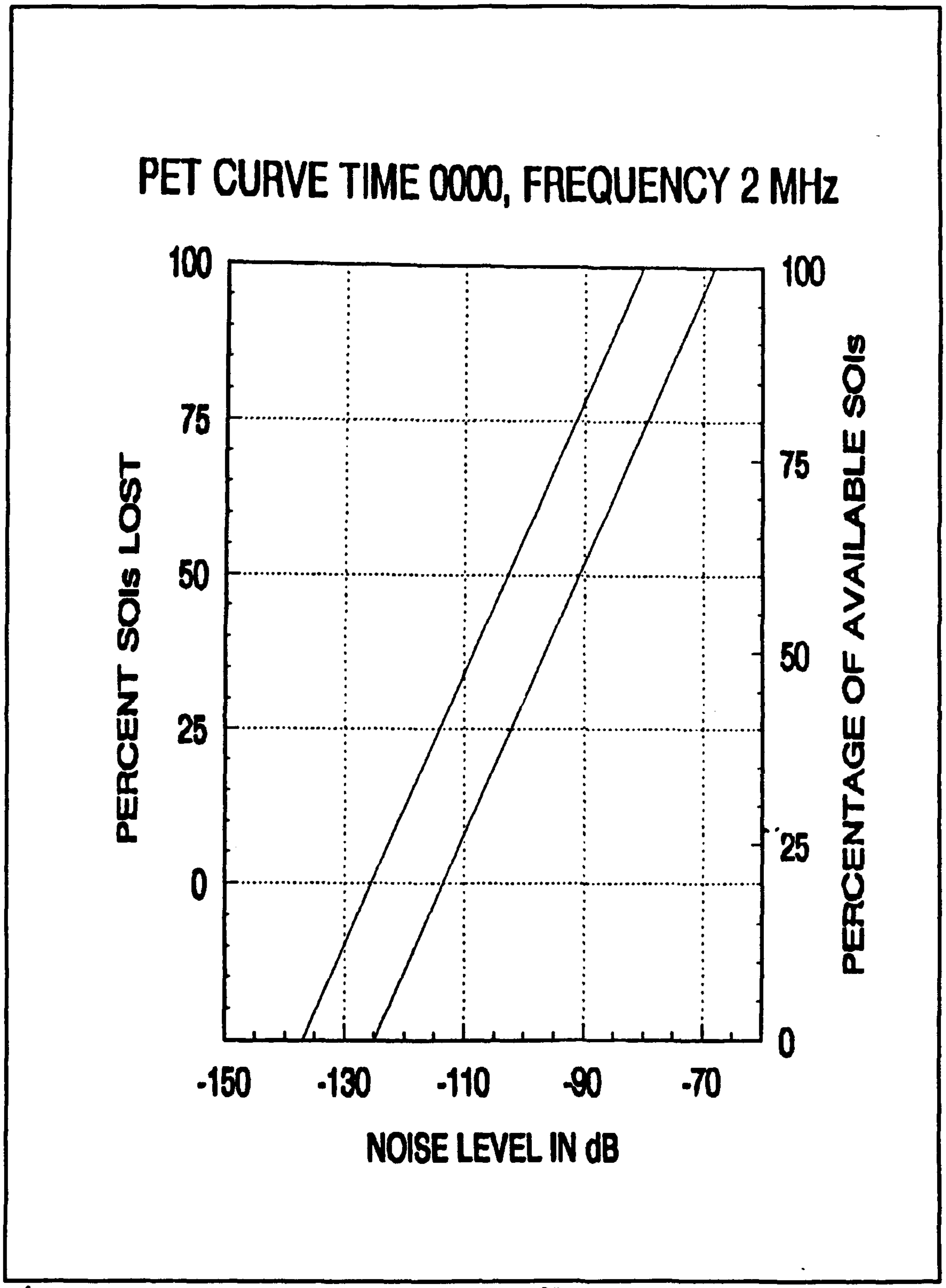

Figure 20 GRAFTOOL PET Plot: 0 dB RFD Loss [Ref. 1]

Performance line is plotted. Figure 21 displays a GRAFTOOL plot with -15 dB RFD loss. [Ref. 1]

Finally, excess noise and either internal or external noise source levels can also be input into the GRAFTOOL PET. This is done through the use of an arrow drawing command. The computation of percentage of SOI lost due to excess noise or noise source level proceeds in the same manner as in the case of manual PET processing. The only difference is that the drawing is being done by the computer. [Ref. 1]

Data can be extracted from the plot and stored for future presentations. This capability is not available in the manual PET methods, including DRAWPERFECT software. After each of the 29 different PET plots for a specific frequency is computed, a plot of percent SOI lost versus frequency for the selected time period can be constructed. This plot uses data stored from each of the 29 different PET plots and looks similar to Figure 22.

The need for an automated PET curve was quite evident because of the large amounts of data collected in a full survey. The GRAFTOOL technique provides some advantages over the manual techniques, but, "even further benefits can be obtained by writing custom computer software." [Ref. 1] Chapter III introduces an automated technique using the high-level language, MATLAB, and Chapter IV compares each of the three PET methods. [Ref. 1]

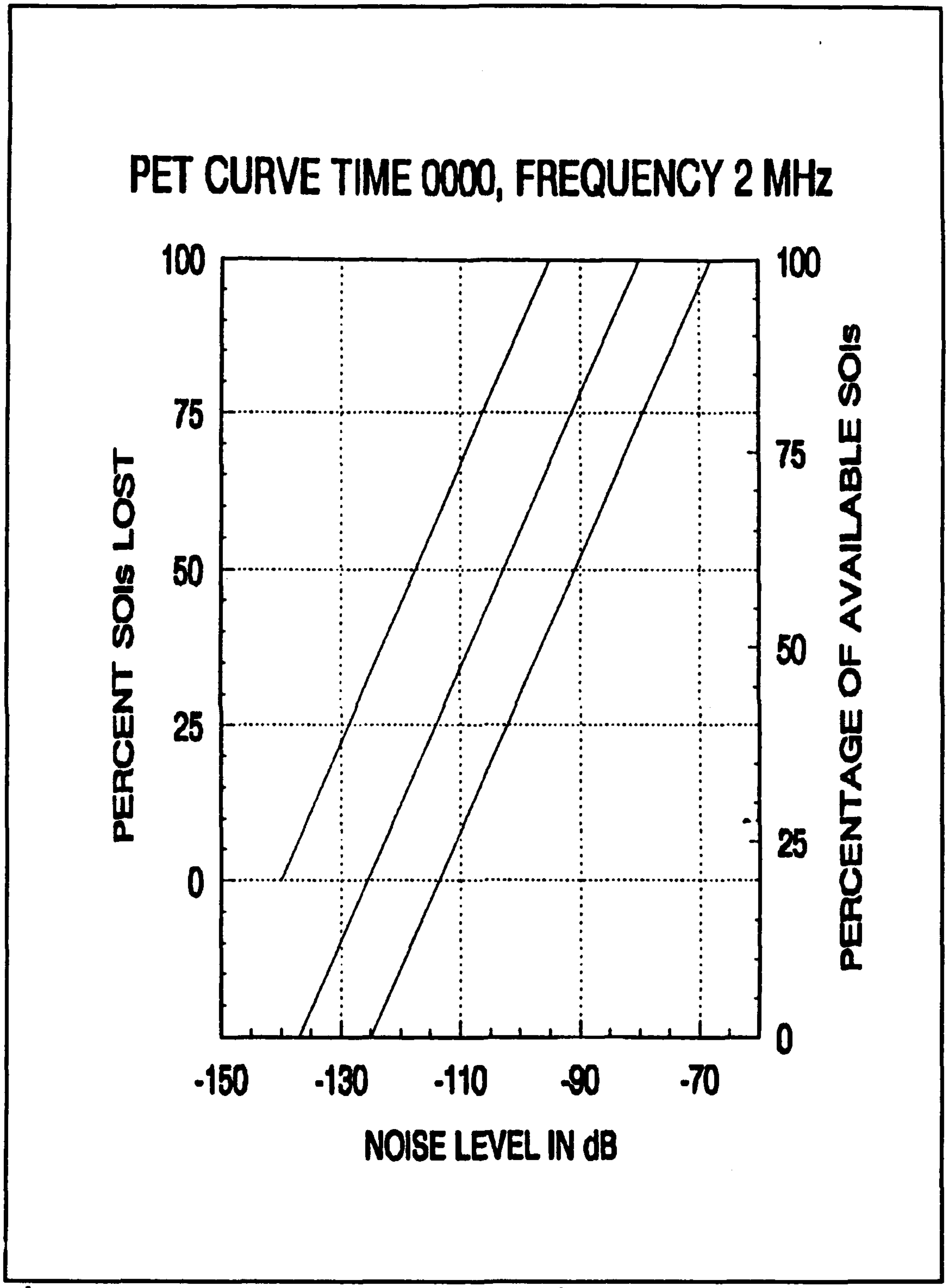

Figure 21 GRAFTOOL PET Plot: 15 dB RFD Loss [Ref. 1]

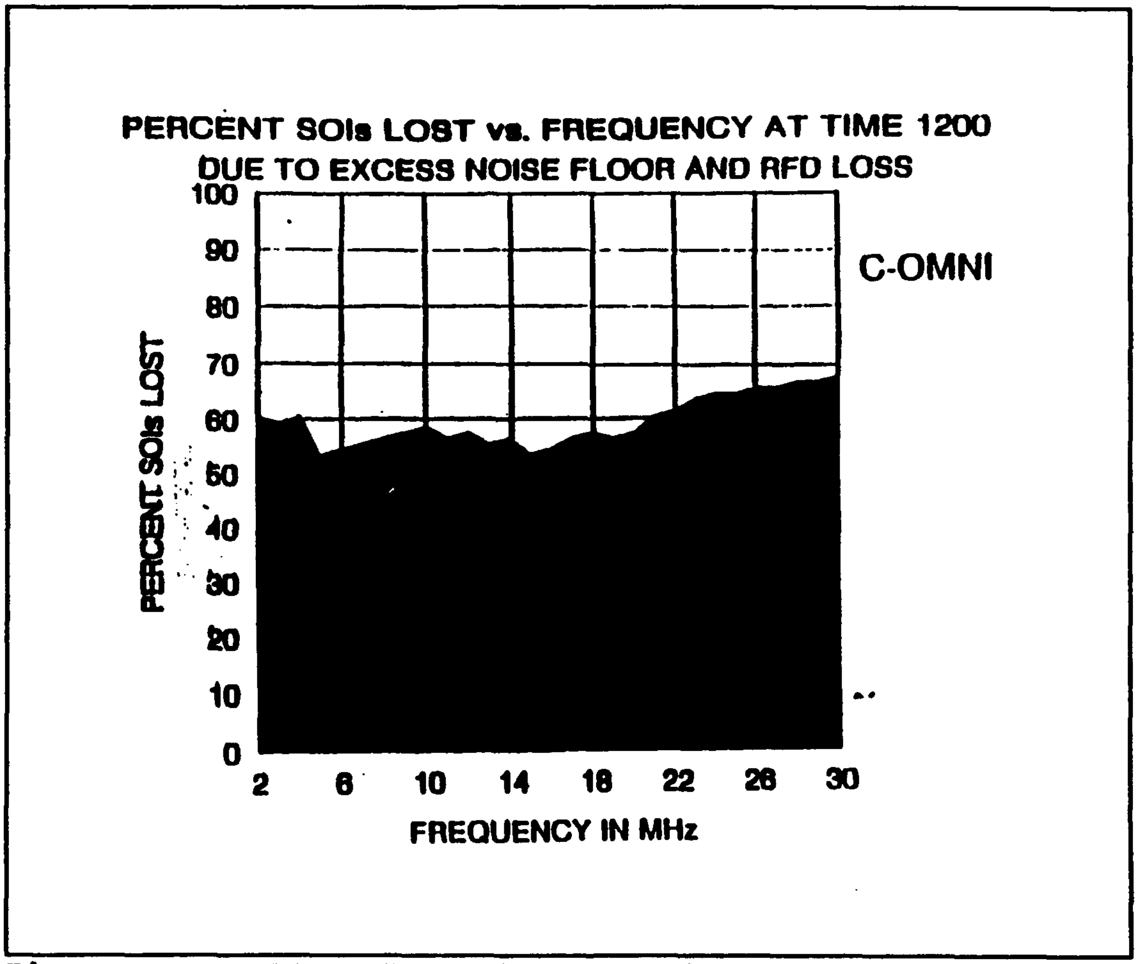

Figure 22 GRAFTOOL Percent SOIs Lost vs. Frequency [Ref. 1]

E. USEFULNESS OF PET ANALYSIS

A major portion of Chapter II has been spent on describing two methods of PET plot construction. It has been shown that the PET can provide information concerning percentage of available signals lost because of RFD losses, excess noise and interfering noise power, whether internal or external. This section further explains the PET's usefulness from the viewpoint of the manager. How can the PET be used to make management decisions?

One of the most important management factors is cost. For the first time, site managers have the ability to make cost-benefit decisions based on quantitative performance analysis. For example, PET plots can be used to determine whether a proposed RFD capable of operating at a reduced noise floor provides the desired increase in SOI reception. PET analysis can be conducted on the proposed system before it is acquired. The results of the SOI available or lost could then be compared to the performance of the system already in place. The manager now has a tool in which he can make a legitimate decision. Is a 5% increase in signals received worth, say, $100,000?

The same cost versus performance trades can be evaluated for repairs, site relocation, or even purchase of land. Would it be cost-effective to make repairs to correct all the power-line noise exterior to the site? Would it be cost effective to purchase land around the site to reduce encroachment, therefore limiting sources of external noise? The answers to these questions can be put on a sound basis with the aid of the quantitative data provided by PET analysis.

The PET can also be used to help evaluate operator performance. As shown earlier, PET analysis can determine whether a specific signal can be intercepted, on the average. If an operator continually claims that the target is nil-heard, the PET can be used to determine, statistically, whether this is expected. This feature could also help

collection managers make decisions on case coverage. If PET analysis continually shows that on bearing 024 and a frequency of 12 MHz that 80% of the SOI are lost, then a collection manager would not want to assign to that site a case operating on that bearing and frequency.

The uses for the PET are quite numerous. However, it must be automated to provide accurate feedback in a timely fashion. The next chapter introduces an automated PET that fills these needs.

III. MATLAB AUTOMATED PET SYSTEM (MAPS)

A complete PET survey of a specific site requires signal and noise measurements as a function of time-of-day, frequency, and bearing. This process requires a vast amount of data processing. Using a frequency range of 2 to 32 MHz with 1 MHz increments, the thirty 12° monitor beams and 24 one-hour time intervals requires that 30 X 31 X 24 = 22,320 PET plots have to be computed for a complete analysis of the performance of a site. Due to the sheer volume of data, automation is required.

In this chapter an automated system using the MATLAB high-level language is introduced. It will be referred to as the "MATLAB Automated PET System" or MAPS throughout the remainder of this thesis. MAPS uses subroutines written in MATLAB to create the various outputs used by SNEP teams. MAPS is capable of producing complete PET plots, signals-lost-vs-frequency plots, signals-lost-vs-bearing plots, signals received-vs-frequency plots, signals-received-vs-bearing plots, and 3-D plots of signals lost vs. frequency and bearing. Time dependence is included when entering maximum signal strength and noise levels into the model. MAPS is capable of computing Percentage of Signals Lost for RFD losses, excess noise floor, and internal or external man-made noise and also determining if a specific signal can be received.

The first goal of this chapter is to review some of the mathematical equations used to produce PET plots in MATLAB. Second, each of the subroutines used in the operation of MAPS is reviewed and, third, an in-depth look at the operation of MAPS is presented.

A. MATHEMATICAL BASIS OF MAPS

In order to use MATLAB, equations had to be developed for the generation of PET plots. A linear approximation to the log-normal distribution was used. Each of the three lines used in PET analysis have two associated equations, one for each of the two axes (Percent SOI Lost and Percent SOI Received). The "Percent SOI Received" axis is similar to an inverted "SOI Available" axis. In order to simplify equation writing, the following abbreviations are used:

- NF   Receiving System Noise Floor (dBm)
- MSS   Maximum Signal Strength (dBm)
- RFD   Losses (dB)
- NS   Noise Level (dBm) (includes internal and external noise and NF + excess noise)

Equations developed for the Percent SOI Lost axis are reviewed first. Refer to Figure 23 for further verification of the following equations.

The first equation establishes the slope of all lines. Figure 23, shows that m1=m2=m3, thus all slopes are equal.

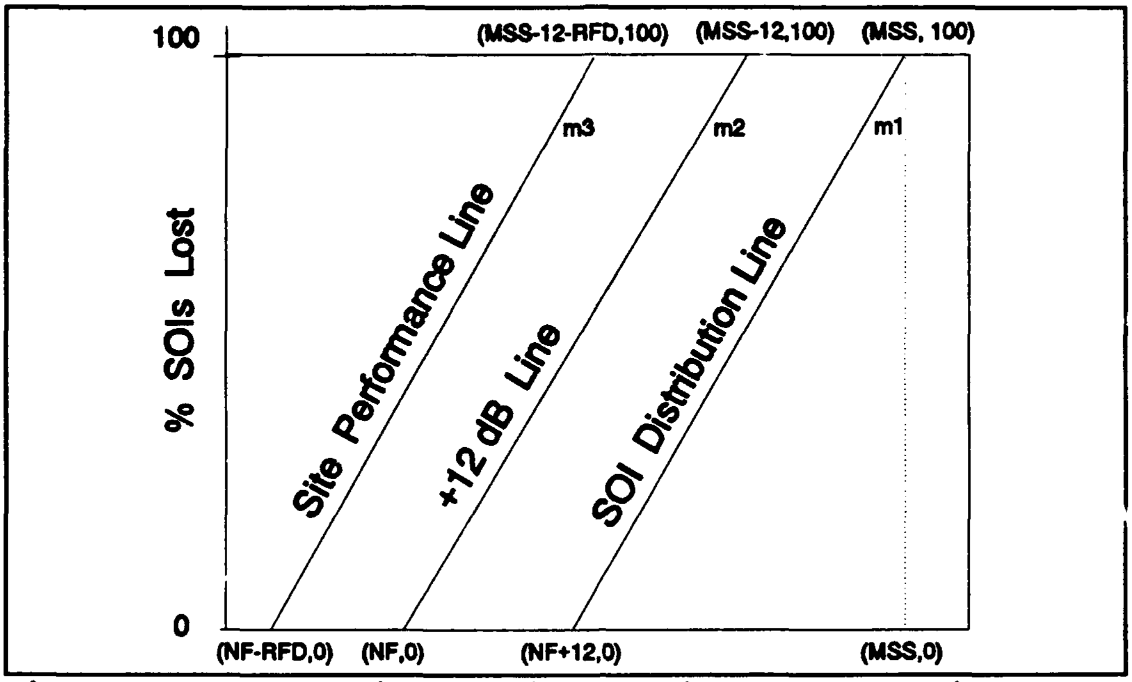

Figure 23 Mathematical Basis for Signals Lost Axis

Using the two points that make up the +12 dB line, the following equation for slope is used:

$$m2=\frac{100-0}{MSS-(12+NF)}=\frac{100}{MSS-12-NF} \quad . \tag{4}$$

Since all three lines have the same slope, m will be used for m1, m2, and m3. After developing the slope equation, the three line equations are construed.

$$y_{SOI}= m(x-12)-m(nf) \tag{5}$$

$$y_{+12dB}= mx-m(nf) \tag{6}$$

$$y_{Site}= m(x+rfd)-m(nf) \tag{7}$$

Finally, after all the above equations are complete, two separate SOI loss equations are required. The first for RFD

losses only, and the second covering losses due to excess noise floor and man-made noise level.

$$loss_{rfd} = m \times rfd \tag{8}$$

$$loss_{noise} = m \times (NS+RFD-NF) \tag{9}$$

With the above equations, MAPS computes "signals lost" for any PET situation.

The next series of equations are used in determining Percent Signals Received. This is a new coordinate axis. It was first used in the SNEP report for NSGA Hanza and is basically the inversion of the "SOI Available" axis. This inversion allows Percent Signals Received to be determined from PET analysis. Figure 24 is provided as a reference for the following equation development.

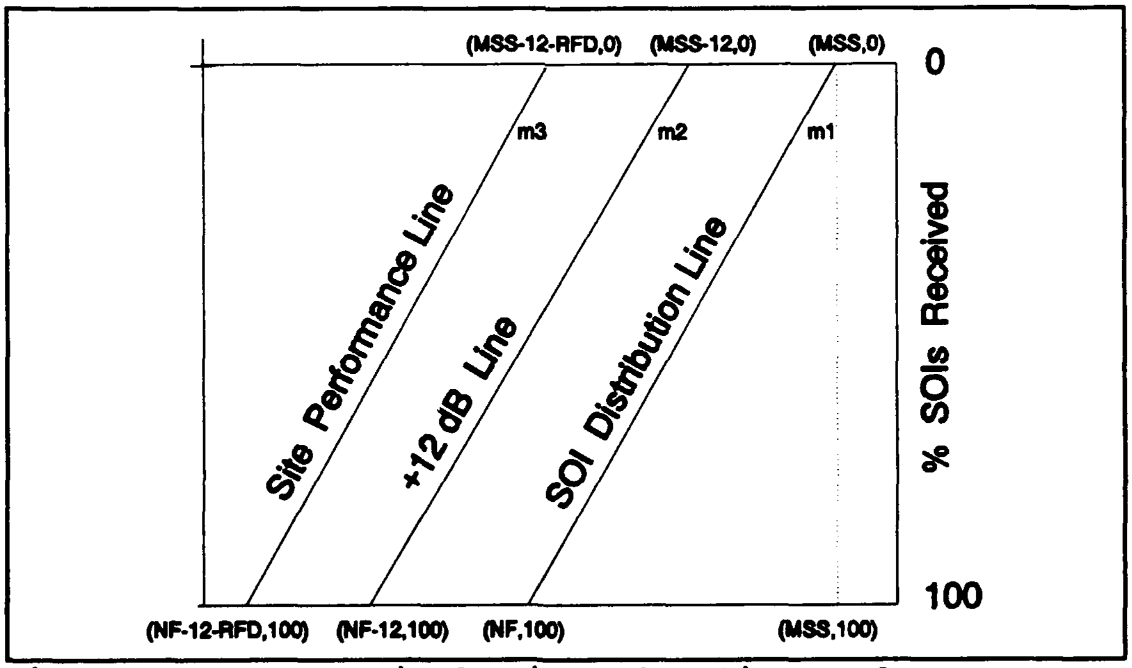

Figure 24 SOIs Received Axis Mathematics [Ref. 5]

Again the slope for all three lines is the same, but the slope for the Percent Received axis is different from that defined above. The slope for the percent received axis is:

$$m=\frac{-100}{MSS-RFD}=\frac{100}{RFD-MSS}\ . \tag{10}$$

Again, after the slope equation is developed, the equations for each of the three lines are deduced.

$$y_{soi}= mx-m(MSS) \tag{11}$$

$$y_{+12dB}= m(x-12)-m(MSS) \tag{12}$$

$$y_{site}= m(x-12-RFD)-m(MSS) \tag{13}$$

Finally, two separate equations for received signals are formed. The first is for RFD losses only and the second for excess noise floor, and internal or external noise sources.

$$rcvd_{rfd}= m(NF+12+RFD)-m(MSS) \tag{14}$$

$$rcvd_{noise}= m(NS+12+RFD)-m(MSS) \tag{15}$$

With the above equations, MAPS is able to compute Percent Signals Received for any PET situation.

After these equations were developed, MATLAB programming was implemented. The following section briefly covers the purpose of each of the subroutines required to run MAPS.

B. MAPS SUBROUTINES

The MAPS system is made up of a main program and 32 subroutines (See Table I). Each of these subroutines are briefly described, with more details included in Appendix A.

Table I MAPS Subroutines

SUBROUTINE	PURPOSE
AVAIL	Determine % SOI Received (Vectors)
AVAILN	Determine % SOI Received (Matrix)
EDEN	Edit system's excess noise file
EDMSS	Edit maximum signal strength files
EDNF	Edit system's noise floor file
EDNOISE	Edit site's noise files
EDRFD	Edit system's RFD gain/loss file
INBRNG	Input desired bearing
INFREQ	Input desired frequency
INITMSS	Set up max signal strength files
INITNOISE	Initialize site noise files
INTIME	Input desired time period
LOADENF	Load system's excess noise file
LOADMSS	Load maximum signal strength file
LOADNFF	Load system's noise floor file
LOADNOISEF	Load site's noise files
LOADRFDF	Load system's RFD gain/loss file
LOSS	Compute % SOI Lost (Vectors)
LOSSN	Compute % SOI Lost (Matrices)
MATEN	Set up system's excess noise file

Table I (cont.) MAPS Subroutines

SUBROUTINE	PURPOSE
MATEN	Set up system's excess noise file
MATRFD	Set up system's RFD gain/loss file
MINSIG	Determine minimum receivable signal strength
OPTOUT	Select desired output option
OUTFILE	Save output for display/print
OUTTIME	Output selected time period
PLOTINT	PLOTPET uses to plot intersections
PLOTPET	Constructs PET plots
REVROW	Reverses matrix row order
SAVENOISE	Save site noise files
SETNF	Set a system's noise floor level
CONVERT.BAS	BASIC program written to convert PROPHET data to MAPS useable data

C. MAPS OPERATION

1. GETTING STARTED

MAPS is a menu-driven program using the high-level language, MATLAB. MAPS requires an IBM compatible computer with a hard drive, a math co-processor, and any version of MATLAB. MAPS is capable of using PCMATLAB, ATMATLAB, or 386MATLAB. Copy the main program and all the subroutines into the directory containing MATLAB using the DOS copy command and MAPS is ready to run.

To start MAPS, enter MATLAB and type "maps". Once this is done, the MAPS main menu should appear on the screen. The main menu is shown in Figure 25. The following sections cover each of the commands listed in the main menu.

1: PET Scratch Pad
2: Compute Signals lost based on stored RFD Losses
3: Compute Signals lost due to stored Excess Noise
4: Compute Signals lost due to stored Noise Sources
5: Signal Reception Determination
6: System Parameter Files Generation/Manipulation
7: Noise Source Files Generation/Manipulation
8: Initialize/Edit Maximum Signal Strength Files

Figure 25 MAPS Main Menu

2. USING THE MAPS SCRATCH PAD

The PET scratch pad or command 1 of the main menu allows a user to compute "back of the envelope" Percent SOI Lost and Percent SOI Received. To use the PET scratch pad, select 1 at the command prompt following the main menu. The user is then prompted to input maximum signal strength (dBm), system noise floor (dBm), RFD Loss(-)/Gain(+) (dB) and noise sources (dBm). If no noise sources are present, then enter

-999 at the noise prompt. If excess noise is to be entered, add the excess noise to the system noise floor and enter this value. For example, if the system noise floor is -125 dBm and the measured excess noise is 2 dB, then enter -123 at the noise prompt. Figure 26 displays the screen just prior to producing the PET plot. Figure 27 displays the output created by command 1. Notice, the right hand corner. The Percent Signals Lost and Percent Signals Received are always listed there. This output varies slightly depending on the input parameters. For example, if there was no noise source present, then the title of the plot states "Signals Lost due to RFD". If a noise source is present, whether it be internal, external noise or excess noise floor, the title of the PET plot reads "Signals Lost due to Noise". Figures 26 and 27 provide examples of a PET plot where no noise sources were present.

```
Command #?  1
Maximum Signal Strength (dBm)?
Noise Floor (dBm)?
RFD Loss/Gain (dB)?
Noise Source Strength (dBm)?
     (Enter -999 if none)
```

Figure 26 MAPS PET Scratch pad (Command 1) Inputs

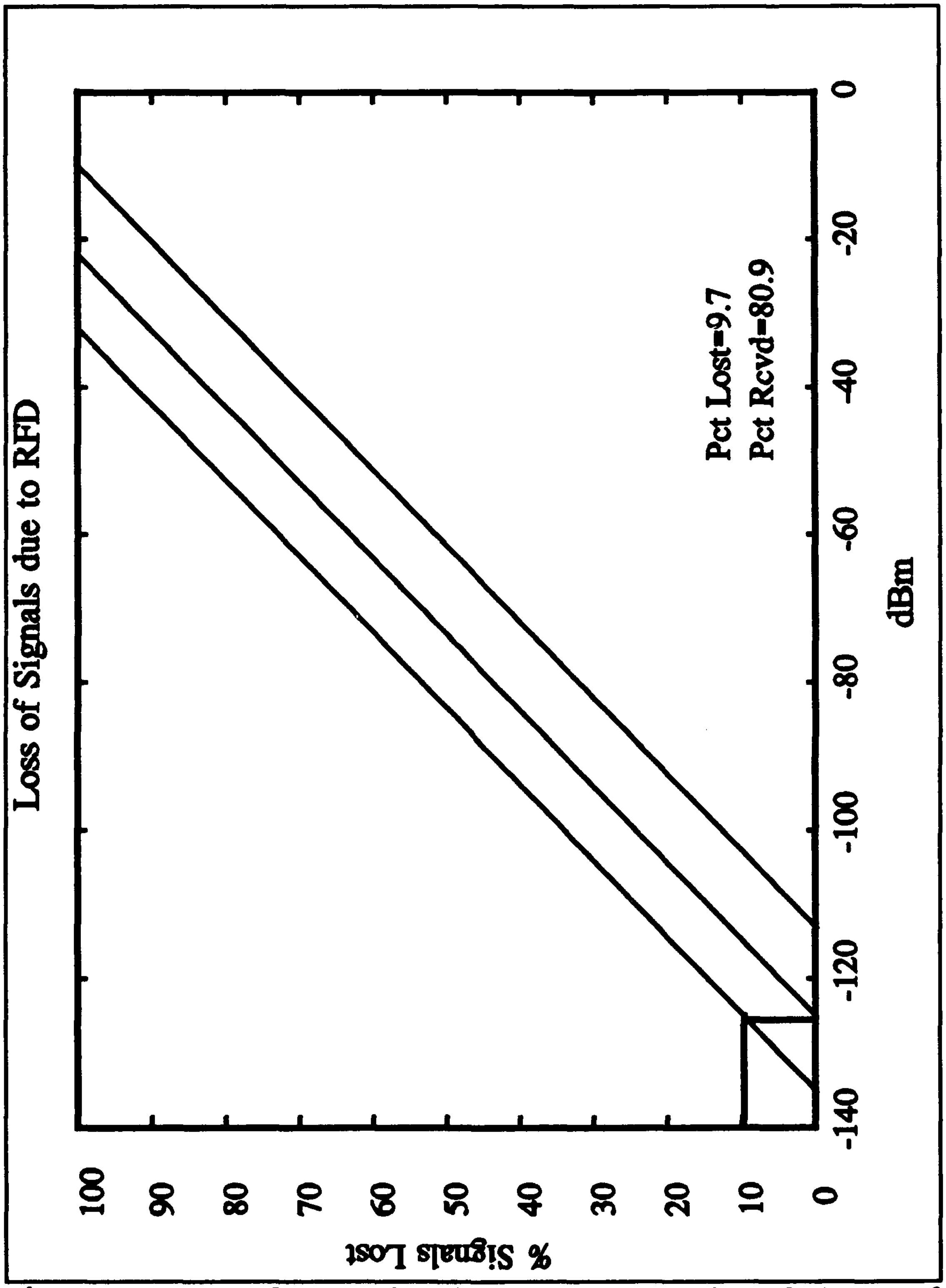

Figure 27 PET Plot produced by MAPS Scratch pad (Command 1)

3. SETTING UP A SYSTEM

Before commands 2 through 4 can be run, system data must be entered into the computer. Command 6 from the main menu allows for the system parameters, system noise floor, RFD loss/gain, and excess noise, to be set up. Type 6 at the main menu prompt to set up a selected system. Figure 28 depicts the screen after 6 was entered.

1: Set up a system
2: View/Edit System RFD Gain/Loss File
3: View/Edit System Excess Noise File
4: View/Edit System Noise Floor

Figure 28 System Setup Screen 1

Select command 1, "Set up a system". The program requests the desired system name. Enter the name in single quotes. For example, if a PET survey was being conducted on a system called SEABAT, the system name would be entered at the prompt as 'SEABAT'. Following the system name, the program requests the system noise floor. Enter this value in dBm. Next the program asks for RFD Gain/Loss values over the frequency range 2-32 MHz. Since the SNEP team currently computes RFD Gain/Loss for only one bearing and assumes this is representative for the remaining bearings, so does the MAPS program. A simple change to the MATRFD subroutine could make the system capable of handling different losses, due to

bearing change. After RFD Gains/Losses have been entered the program seeks data concerning system excess noise. Once again MAPS requests data at integer frequencies in the range 2-32 MHz and does not consider variations due to bearing. If desired in the future, this restriction can easily be lifted by code modification.

Any number of systems can be set up. Each system that is processed has three associated files. The system noise floor file is denoted by nfsystem.mat. The system RFD gain/loss file is denoted by rfdsystem.mat and the system excess noise file is denoted by ensystem.mat. For example, after the SEABAT system was set up, nfseabat.mat, rfdseabat.mat, and enseabat.mat are placed in the MATLAB directory. The noise floor file holds the system noise floor as its sole entry. The RFD gain/loss and excess noise files both are 31 X 31 matrices. The columns cover the frequency range 2-32 MHz and the rows cover each of the thirty 12° monitor beams. The MAPS system was programmed to handle different values due to change in bearing. Since, the SNEP team does not compute the RFD loss and excess noise parameters for each bearing, the data input routines used to set up these files were designed to operate with the current requirements. Because of this, changes due to bearing are only reflected by noise signal strength files and maximum signal strength files. A simple modification to the RFD loss and excess noise file creation subroutines would alleviate this restriction.

After setting up a system, a user can compute percent SOI lost/received due to RFD losses and excess noise for that system. However, percent SOI lost/received due to internal/external noise sources cannot be computed because these files have not yet been established.

4. SETTING UP INTERNAL/EXTERNAL NOISE FILES

Following the discovery of interfering noise sources, the MAPS noise files should be set up to permit computations of Percent Signals Lost due to interfering noise source levels. Since internal/external noise sources can vary with bearing, frequency and time, MAPS stores noise sources in six separate files. Each of the files contains a 31 X 31 matrix identical in form to the matrices held in the system RFD gain/loss and system excess noise files. The matrix includes the variation with frequency and bearing. The six different files allows for the variation with time. Each noise file contains noise power in dBm for a four-hour period. The relationship between files and time periods is as follows:

File 1: 0600 - 1000 local

File 2: 1000 - 1400 local

File 3: 1400 - 1800 local

File 4: 1800 - 2200 local

File 5: 2200 - 0200 local

File 6: 0200 - 0600 local

These four hour increments were chosen based on a review of PROPHET field strength predictions and past SNEP team surveys.

To establish a site's noise files, they must first be initialized by setting each entry to the default value of -999 dBm. This is accomplished through main menu command 7, "Noise File Generation/Manipulation". After entering command 7, the computer displays the menu shown in Figure 29. Select 1 to initialize the six noise files.

```
1:  Initialize Noise Source Files
2:  Edit/View Noise Source Files

Enter Command?  1

This function initializes the six Noise Source
Files.  It will erase the current Noise files.
Type ctrl-c to quit, or return to continue.
```

Figure 29 MAPS Noise File Initialization

The next step is to enter discovered noise sources. Select Command 7 from the main menu, then select command 2, "View/Edit Noise Signal Files", from the menu shown by Figure 29. Three options are displayed on the screen. Select command 1, the "add" option. The system then requests the desired time period, followed by the whole-MHz frequency

affected, bearing, and signal strength of the interfering noise source. Follow the on-screen instructions to continue entering sources or to quit and return to the main menu.

5. MAXIMUM SIGNAL STRENGTH FILES

Maximum signal strength files are very important to PET processing. However, maximum signal strength is a very difficult parameter to measure or predict because of variations with time, frequency and bearing. Normally, known sources are predicted using software such as PROPHET, but quite frequently there are not enough known signals located over the entire 360° azimuth. Currently, the SNEP team selects a few known transmitter sites, computes maximum signal strength for those signals, and ensures that RFD losses and excess noise are computed for each of the known bearings. In order to make a complete PET survey all bearings should be covered, a very time-intensive effort.

Currently, MAPS has the capability to cover all bearings if the measurements are made. Since this is not usually the case, the initialization of maximum signal strength files treats each bearing equally. Maximum signal strength estimations differing in bearing can be entered into the existing maximum strength files by using the editing routine described later. There are a total of six maximum signal strength files. Each file is made up of maximum signal strength in dBm versus frequency and bearing. Each file is

also tagged with a time period. The time periods chosen are the same used for the interfering noise level files.

a. INITIALIZATION OF MAXIMUM SIGNAL STRENGTH FILES

MAPS is setup to take ASCII files or MATLAB ".MAT" files containing a 31 X 31 bearing and frequency matrix. However, when PROPHET is used the data is not in a format readable by MAPS. PROPHET outputs a 24 X 20 time-and-frequency matrix which does not include bearing information. Furthermore, in order to use PROPHET data for MAPS input, two different field strength output files are needed because PROPHET calculations are restricted to 20 different frequencies per output matrix. This restriction is due to the fact that PROPHET is limited to 80 columns: 60 columns for field strength calculations for each of the 20 different frequencies and 20 columns for miscellaneous information. To accomodate PROPHET, use a minimum frequency of 2 MHz and a maximum frequency of 20 MHz for the first PROPHET field strength file. PROPHET provides field strengths spanning 1 to 20 MHz in single MHz steps. Use a minimum frequency of 21 MHz and a maximum frequency of 32 MHz for the second field strength file. PROPHET then responds by providing field strength for 2 to 40 MHz in two MHz steps. The BASIC program CONVER.BAS was written to convert the PROPHET field strength files into maximum field strength files, one containing a 24 X 20 time-and-frequency matrix and the second containing a 24

X 11 time-and-frequency matrix. Since PROPHET determines the lowest useable frequency (LUF) and the maximum useable frequency (MUF), it does not always provide data for each whole MHz from 2 to 32 MHz. The CONVER.BAS routine, therefore, treats these "holes" as the lowest signal level available (-99 dBμv) through PROPHET. The MAPS subroutine INITMSS is used to transform the converted PROPHET data files into the six time differentiated data files, each containing a 31 X 31 bearing-and-frequency matrix, required by MAPS.

The subroutine INITMSS creates the six different time period files, converts the PROPHET field strengths from dBμv to dBm, and places them in a bearing and frequency matrix. The conversion from dBμv to dBm is accomplished by adding -107, based on a 50-ohm matched impedance. Because of this, a signal level of -206 dBm becomes the "flag" for unavailable maximum signal strength data. Since variation with respect to bearing is not currently the major concern of SNEP teams, the INITMSS subroutine places the maximum signal strength per frequency in each of the thirty-one bearing slots. MAPS, however, allows inputs to vary with bearing. It does this through the subroutine EDMSS, designed to allow editing of maximum signal strength data files.

b. EDITING MAXIMUM SIGNAL STRENGTH FILES

The ability to edit the maximum signal strength files created for MAPS allows the user to take into

consideration variations due to bearing. When another transmitter site is identified and the maximum signal strength predicted, the user can load this information into the bearing slots associated with the received energy. Each of the six different time-period oriented files are adjusted to reflect the new information. This is done through the use of the EDMSS subroutine. The user is prompted to enter the converted PROPHET data filename and the bearing to the transmitter. The subroutine then adjusts each of the six MAPS files as necessary.

6. VIEW/EDIT SYSTEM PARAMETER FILES

a. VIEW/EDIT RFD GAIN/LOSS FILE

MAPS allows for the editing and the display of a system's RFD gain/loss file. From the main menu, enter command 6, "System Generation/Manipulation". The screen should resemble Figure 28. Enter command 2, "View/Edit System RFD Gain/Loss File". The system then displays all the available RFD files of previously established systems. If a file is not listed, it has not been setup. Do not try to edit or view a system if it has not yet been set up. One can exit the program by pressing ctrl-c. The choices of "Replace RFD Gain/Loss" values and "View RFD Gain/Loss File" should also be present on the screen. The first allows for the editing of the chosen system RFD gain/loss file. The actual editing is self-explanatory, so that few difficulties should arise. The

second choice allows for the display of a system's RFD gain/loss file. The two plots available to the user are shown by Figure 30. The top plot presents the variation of system RFD gain/loss with frequency and the lower plot displays the variation with bearing. Bearing variation is not entered into PET data yet, so all plots that vary with bearing will be comprised of straight lines.

b. VIEW/EDIT SYSTEM EXCESS NOISE FILE

MAPS allows for a system's excess noise file to be exhibited or edited. From the main menu enter command 6, system generation/manipulation. From the menu displayed by Figure 28 select command 3, "View/Edit System Excess Noise File". Excess noise filenames for previously setup systems are shown. Excess noise files that have not been set up cannot be viewed or edited. The two choices of either "view" or "replace" excess noise values are also displayed. Editing of the file is self-explanatory and the available plots of a system's excess noise file are shown in Figure 31. Since excess noise data is not entered with bearing variation, the vs. bearing plots will be comprised of straight lines.

c. VIEW/EDIT SYSTEM NOISE FLOOR FILE

MAPS allows the user to view or edit the existing system noise floor. This is accomplished through main menu command 6, followed by command 4 from the menu displayed by Figure 28. Available systems are displayed. Choose the

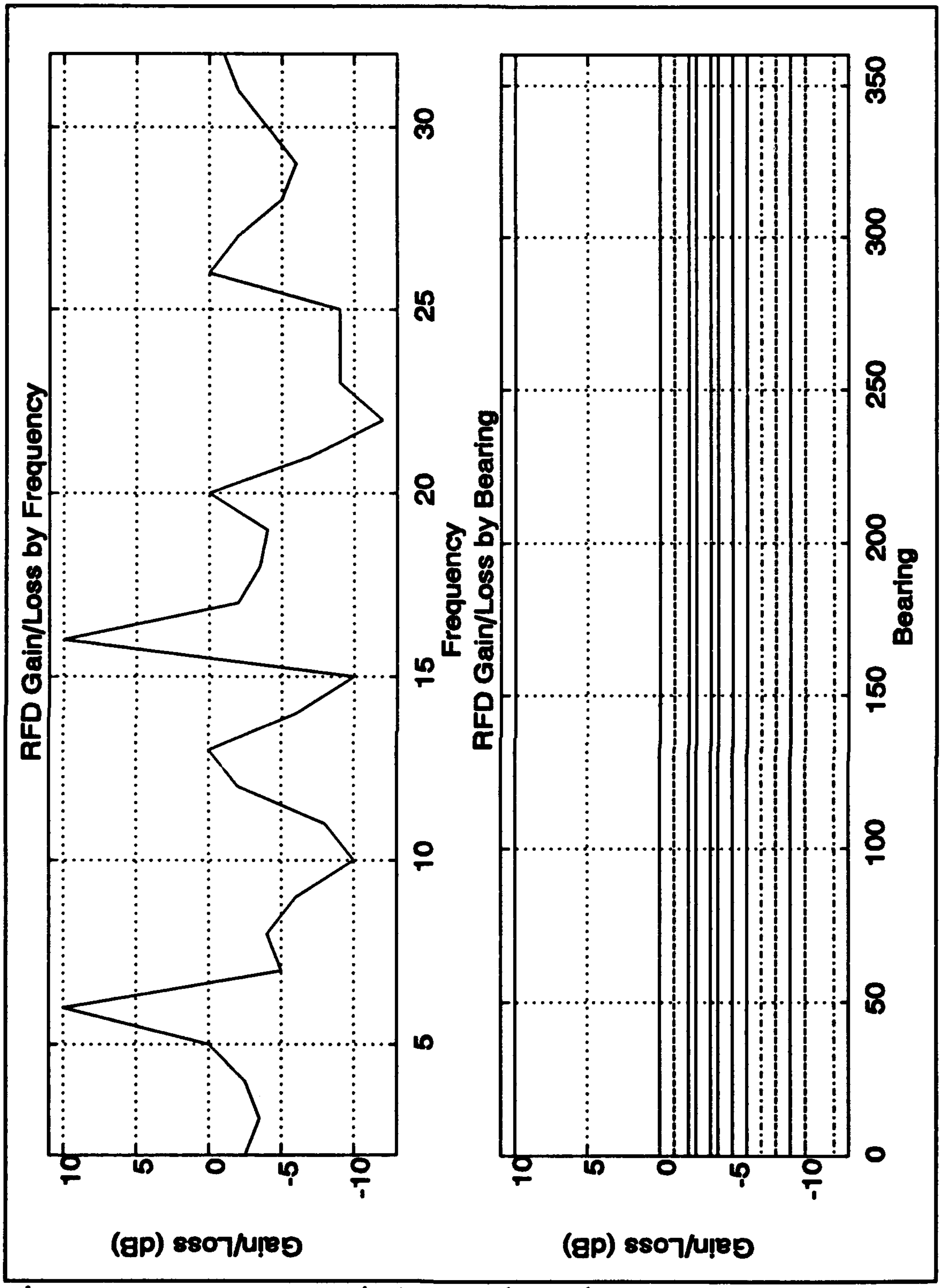

Figure 30 MAPS RFD Gain/Loss File View

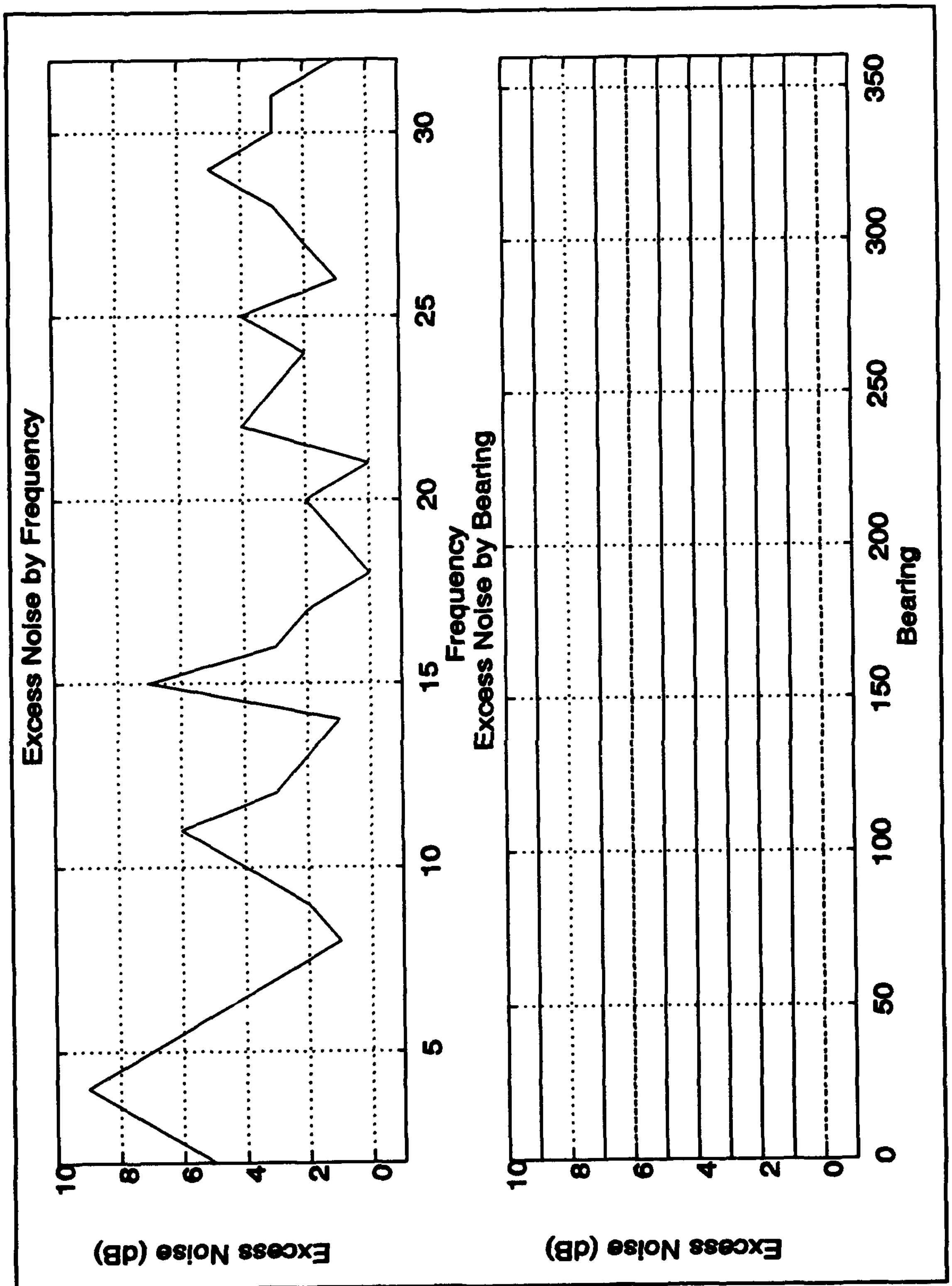

Figure 31 MAPS Excess Noise File View

desired system. The noise floor value is exhibited and the user is prompted to change or not to change the value.

7. VIEW/EDIT INTERNAL/EXTERNAL NOISE FILES

Adding new noise sources was already explained in Section 4. Therefore, this section concentrates on deleting noise sources and viewing the noise source level files.

Deletion of noise sources is accomplished by setting their values to the default value of -999 dBm. This is accomplished through command 7 of the main menu shown in Figure 25. Next, choose the "delete" option. The user is queried about the desired frequency and bearing of the noise source about to be initialized followed by an "are you sure you want to delete" system stop. Follow the on-screen instructions to continue to delete sources.

Viewing noise files is accomplished through command 7 of the main menu. Select the "view" option next. MAPS then presents two plots, shown by Figure 32, of the selected noise file. Noise sources are entered with bearing variation, therefore the versus bearing plot shown in Figure 32 is not solely made up of straight lines.

8. COMPUTING SIGNALS-LOST USING STORED PARAMETERS

In order to complete a site PET survey, MAPS requires that the system and noise files be generated. MAPS is capable of computing Percent SOI Lost/Received based on RFD losses, excess noise floor, and internal/external noise sources.

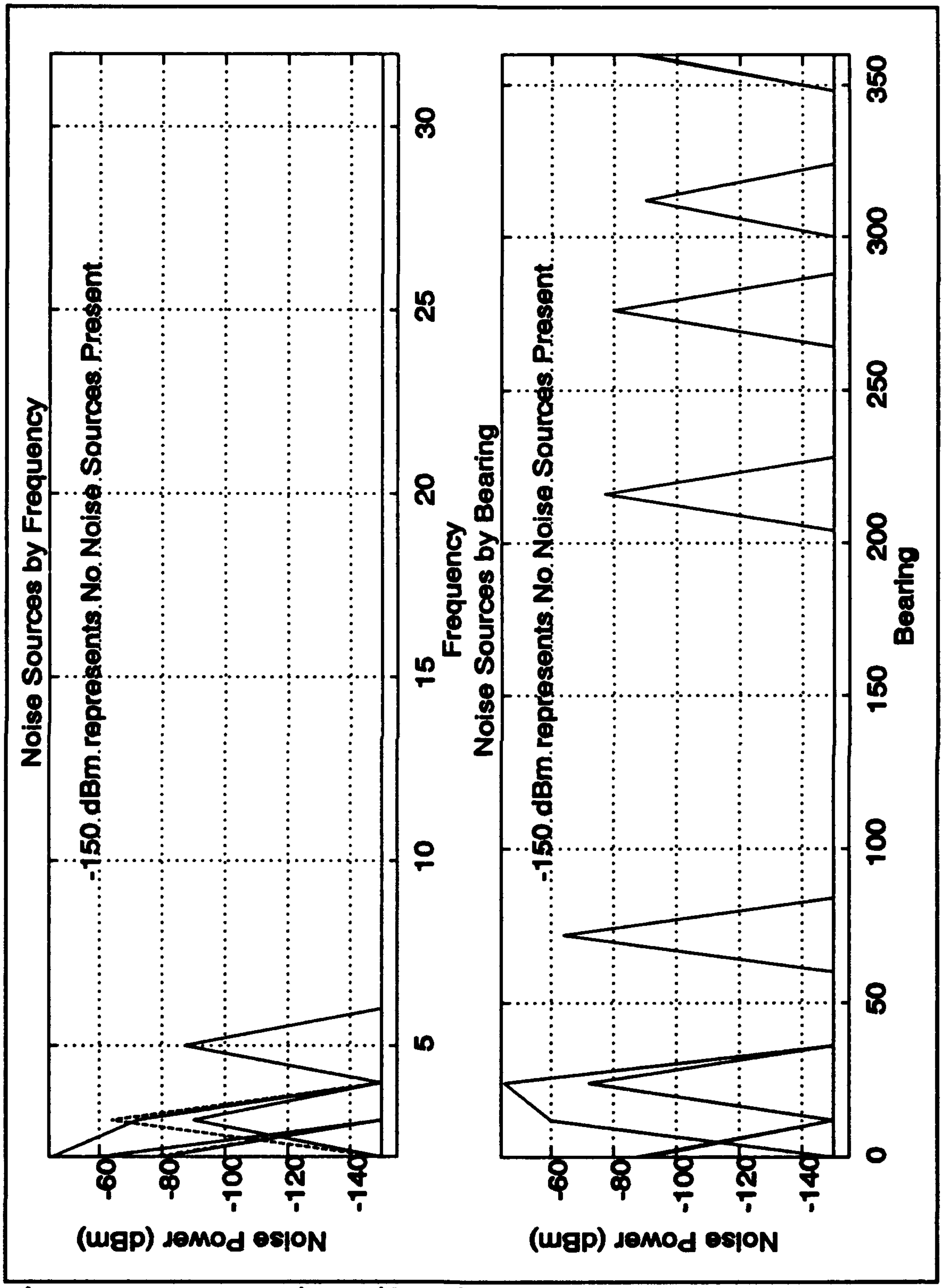

Figure 32 MAPS Noise File View

Besides being able to provide these computations, MAPS is able to provide four different choices for output. These four output options are:

- A PET plot based on a single bearing and single frequency.
- Two subplots, Percent Signals Lost vs. frequency, and Percent Signals Received vs. frequency.
- Two subplots, Percent Signals Lost vs. bearing, and Percent Signals Received vs. bearing.
- Two subplots, a 3-dimensional representation of Percent Signals Lost vs. frequency and bearing, and a contour plot of the 3D diagram.

Examples of each are shown in Figures 33 through 36. In these plots, loss of signals due to atmospherics or lack of maximum strength signal data is displayed by Percent Signals Lost and Percent Signals Received set to zero.

a. COMPUTING SIGNALS-LOST BASED ON RFD LOSS FILES

Before attempting to use this command, ensure that the desired system has been set up. MAPS allows users to compute Percent Signals Lost and Percent Signals Received solely due to system RFD losses. This is accomplished through main menu command 2.

After entering command 2, the system requests the name of the system to evaluate. All available system RFD loss files are displayed in the format RFDsystem.mat. Enter the system name in single quotes. Do not include the prefix RFD or the suffix .mat. Select the desired time period and the desired output. MAPS then exhibits the chosen output.

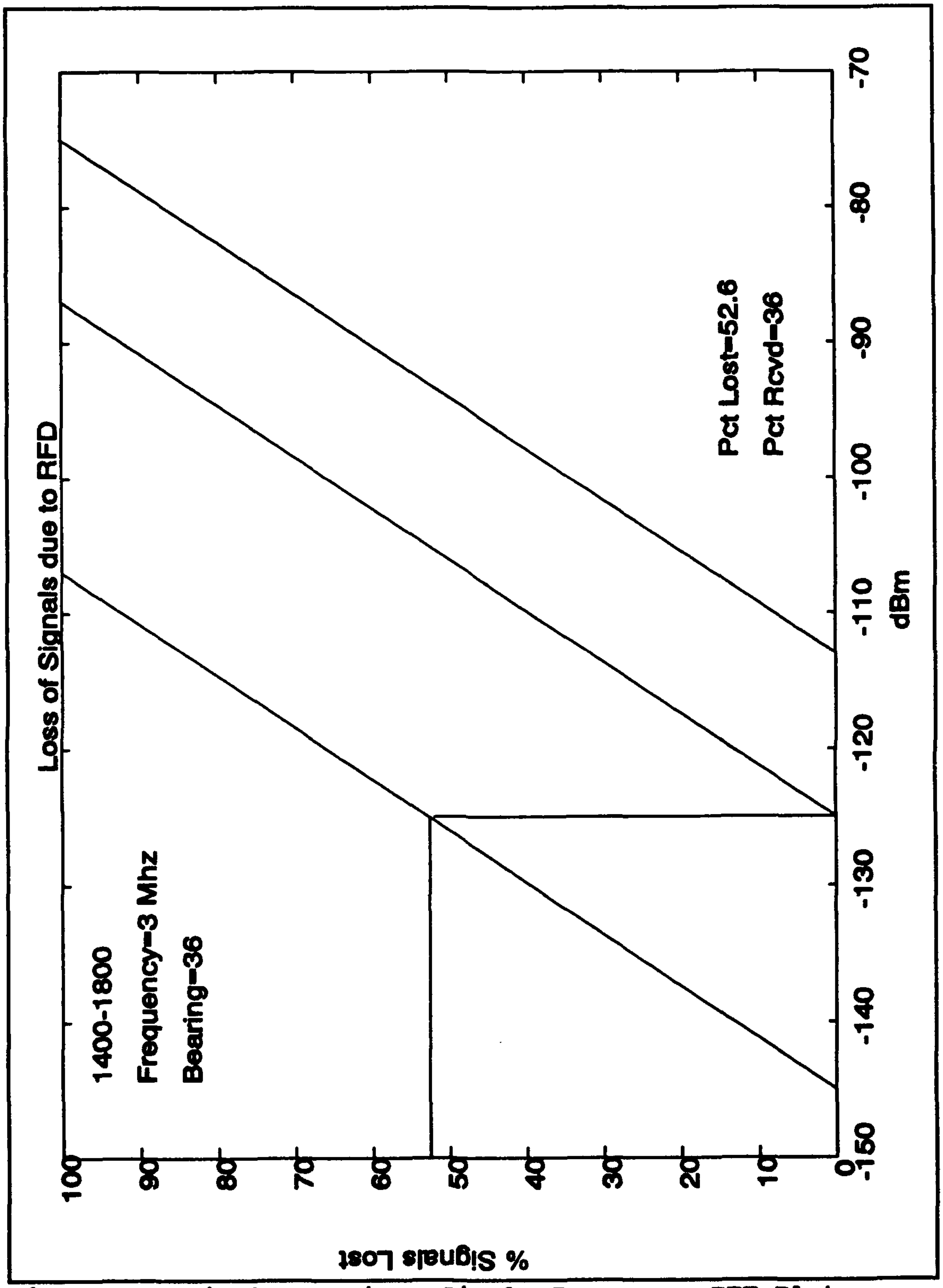

Figure 33 Single Bearing, Single Frequency PET Plot

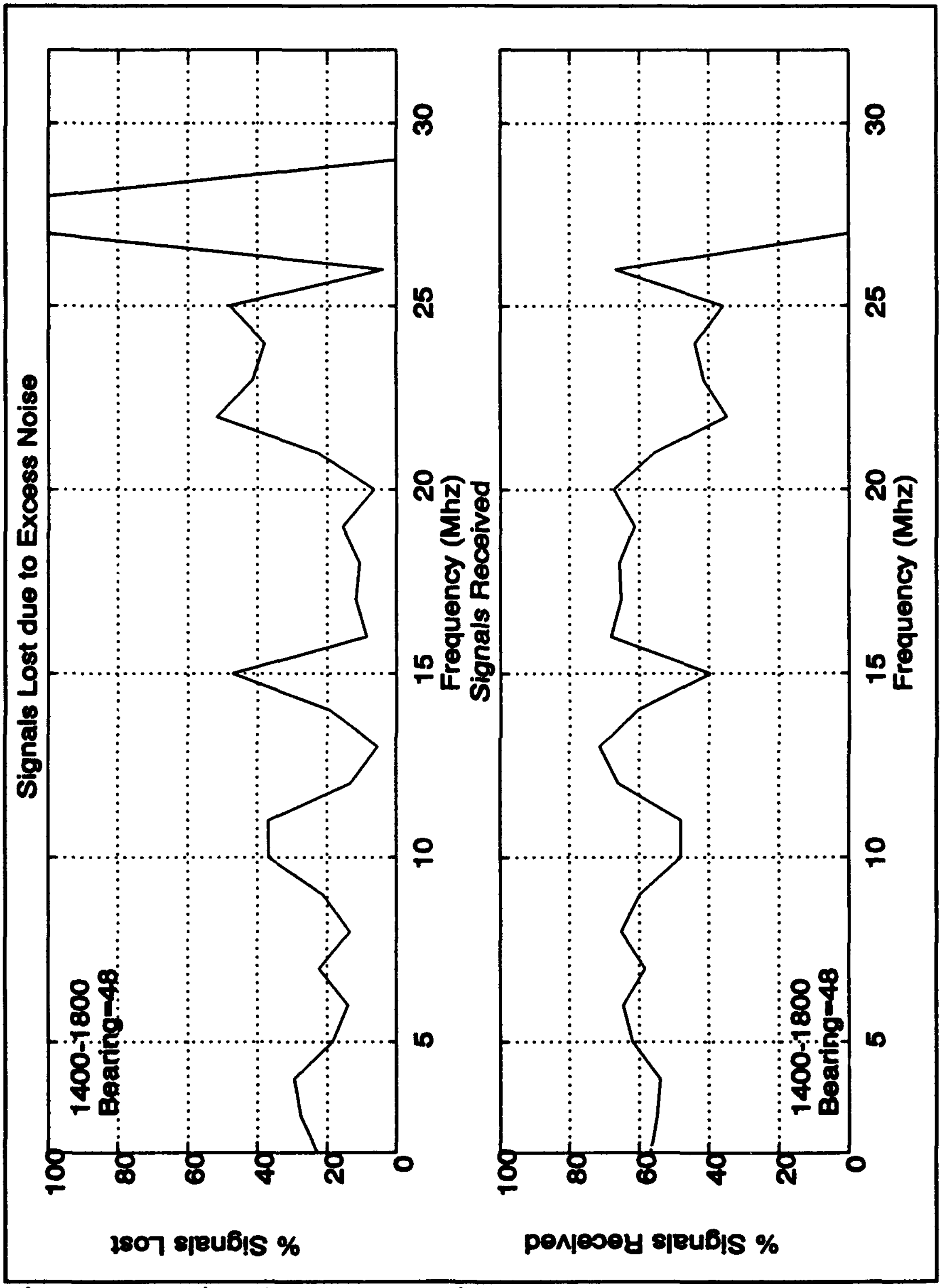

Figure 34 Signals Lost/Received vs. Frequency Plots

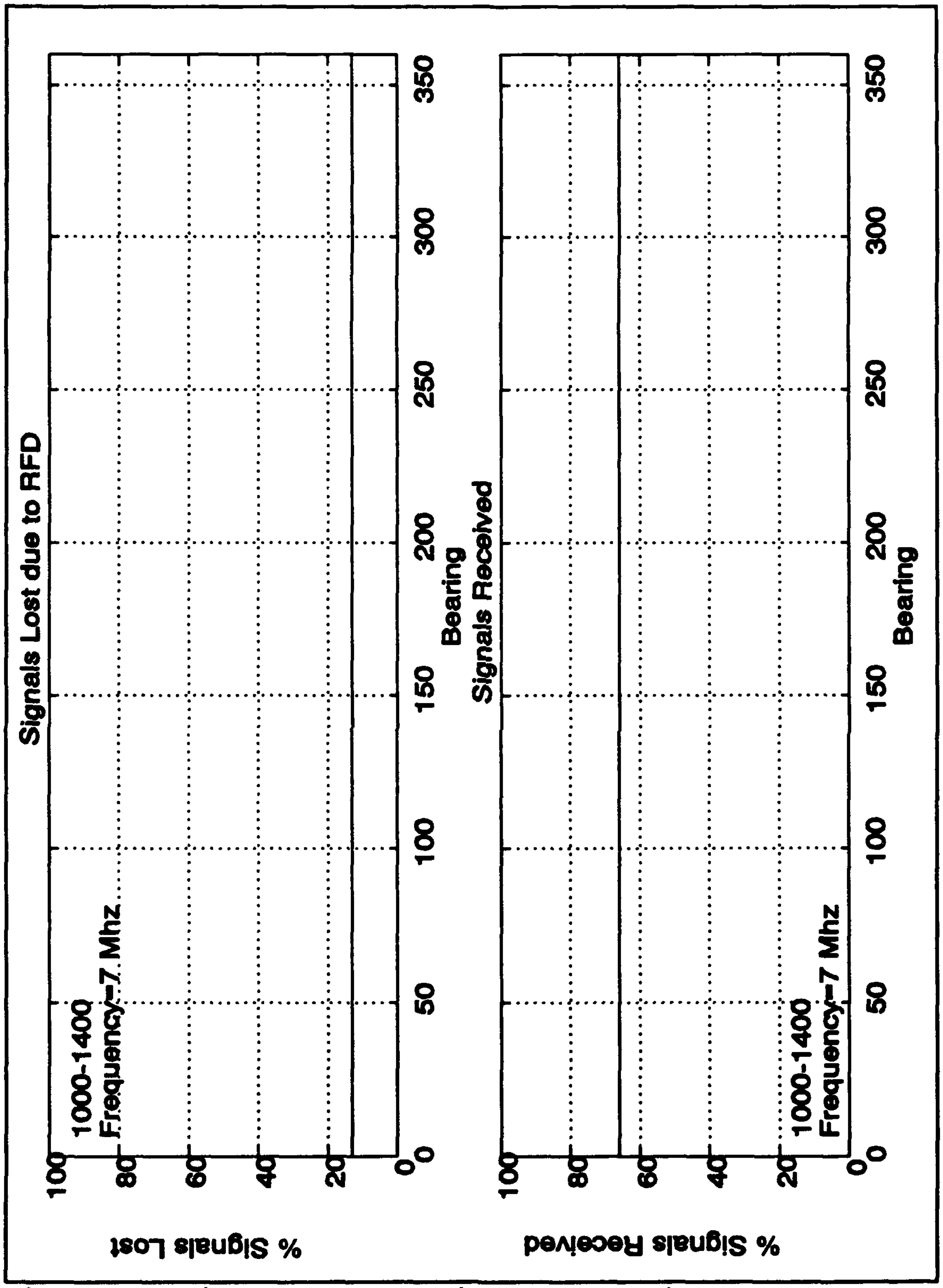

Figure 35 Signals Lost/Received vs. Bearing Plots

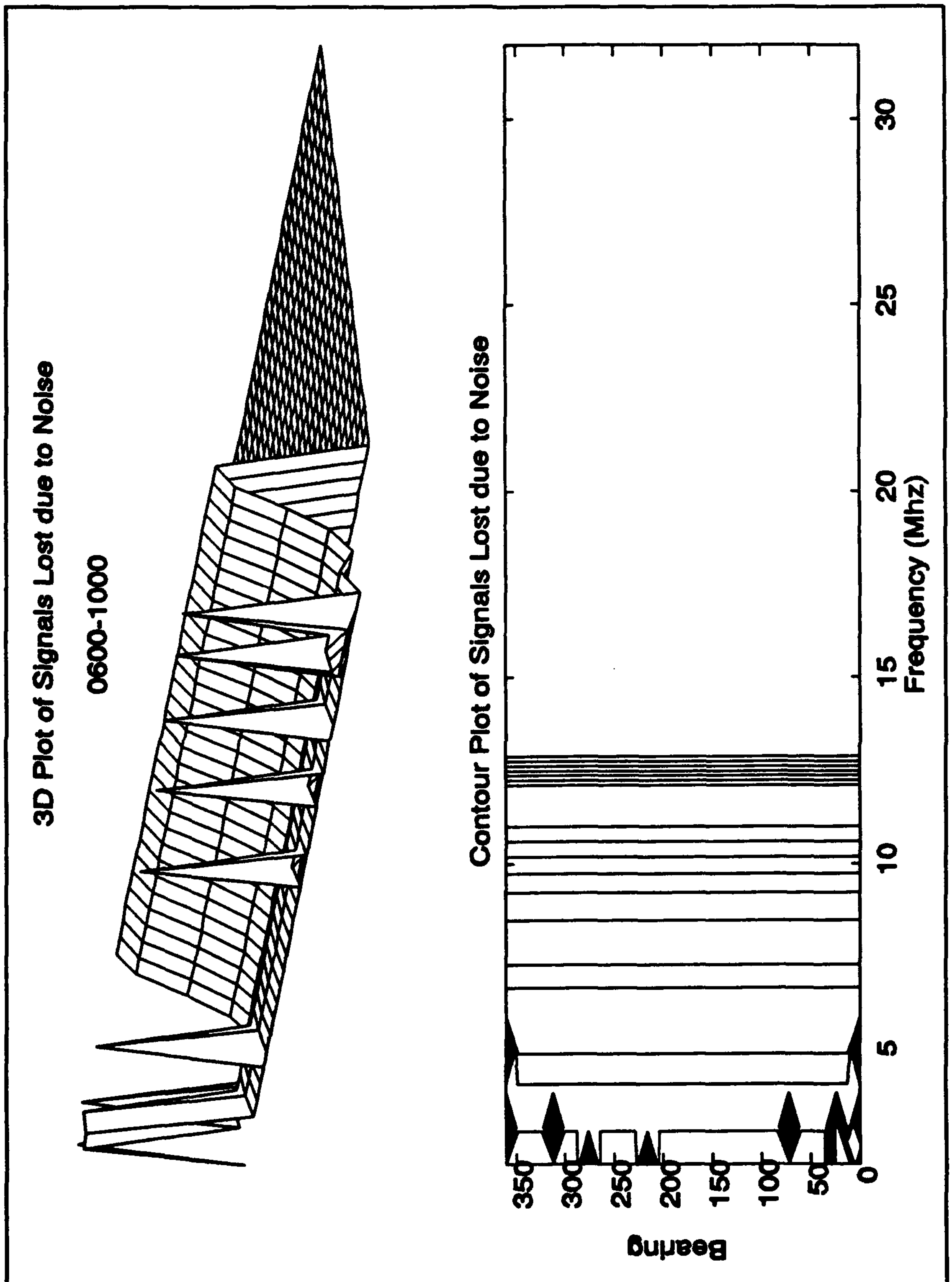

Figure 36 3D and Contour Plots of Signals Lost vs. Frequency and Bearing

b. COMPUTING SIGNALS-LOST BASED ON EXCESS NOISE FILES

Before using this command, ensure the desired system has been set up. MAPS computes Percent Signals Lost and Percent Signals Received due to excess noise through main menu command 3. When making these calculations, MAPS takes into consideration RFD losses as well as excess noise. If there is no excess noise, MAPS computes Percent Signals Lost due to RFD losses.

After entering command 3, again one is requested to enter the system name in single quotes. Do not enter the prefix "en" or the suffix ".mat". Select the desired time period and output. MAPS then produces and displays this output.

c. COMPUTING SIGNALS-LOST DUE TO NOISE SOURCES

MAPS is capable of computing Percent Signals Lost and received due to internal or external noise sources. Again, ensure that the desired system is set up before attempting to run this command. Also, ensure the site's six noise files have been, at least, initialized. Percent Signals Lost due to noise sources is carried out through main menu command 4.

After entering command 4, the user is requested to enter the desired system in single quotes. All systems available are listed. Do not enter the prefix "RFD" or the

suffix ".mat". Select the preferred time period and output selection. MAPS then displays the selected output.

9. DETERMINING SIGNAL RECEPTION CAPABILITY

MAPS is capable of determining, on the average, whether a receiving system is able to intercept a specific SOI. In order to use this command, the expected value of the received signal strength must be known or have been predicted by software such as, PROPHET. Reception capability is accomplished through main menu command 5. After entering command 5, available systems are displayed. The user must enter the desired system in single quotes. Next, the user must choose one of the six time periods: 0000-0400, 0400-0800, 0800-1200, 1200-1600, 1600-2000 or 2000-2400.

1: Initialize new maximum signal strength files
2: Edit current maximum signal strength files
3: View current maximum signal strength files

Figure 37 Reception Determination Menu

After selecting the time period, the menu shown by Figure 37 is displayed. MAPS provides a value or plot of the

minimum received signal strength that, on the average, can be intercepted by the selected system. Command 1, "Frequency and Bearing Known", allows the user to input a specific frequency and bearing and determine the minimum signal strength that can be received. Command 2, "Frequency Known", provides a plot of minimum signal strength versus bearing for the known frequency. Command 3, "Bearing Known", provides a similar plot versus frequency for a prescribed bearing. Plots resulting from commands 2 and 3 are shown in Figures 38 and 39. After obtaining the chosen output, the predicted received signal strength for a specific signal is compared to the minimum receiveable signal strength to determine whether or not the system is capable of receiving the signal.

10. PRODUCING HARD COPY MAPS OUTPUT

Besides allowing a user to view the MAPS output on the monitor display, hard-copies of all output can be obtained. This can be accomplished by two different methods. In the first method, when a plot is displayed on the screen, the "Print Screen" key can be used to obtain hard copy. The second, preferred, method is incorporated into the MAPS program and makes use of the MATLAB command, "meta". In this latter method, after a plot is displayed and the "return" key hit, MAPS asks whether hard copy output is wanted. If so, the user must input a filename in single quotes. For example, if the filename 'seabag' was entered, the plot previously on the

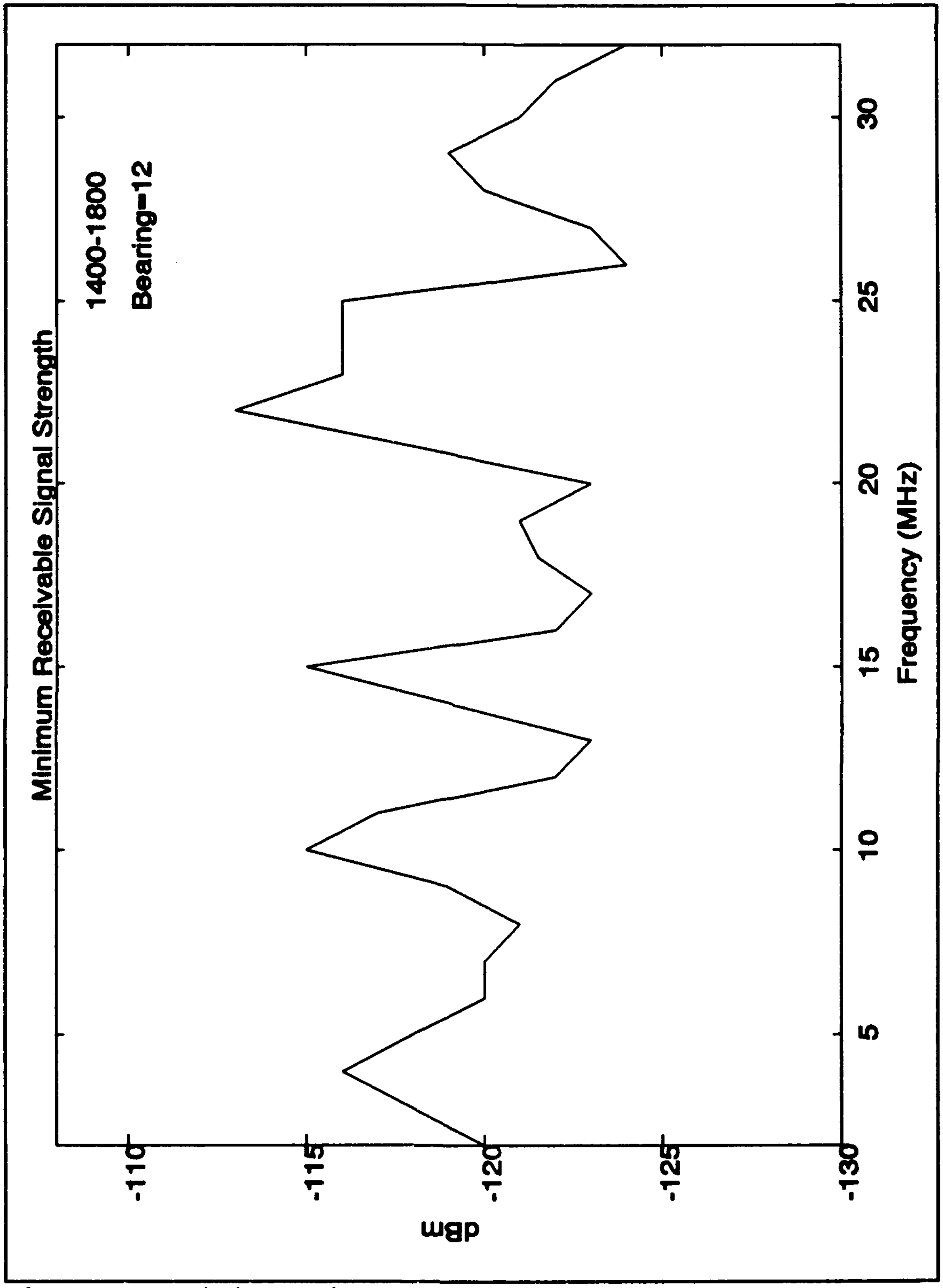

Figure 38 Minimum Signal Strength vs. Frequency Plot

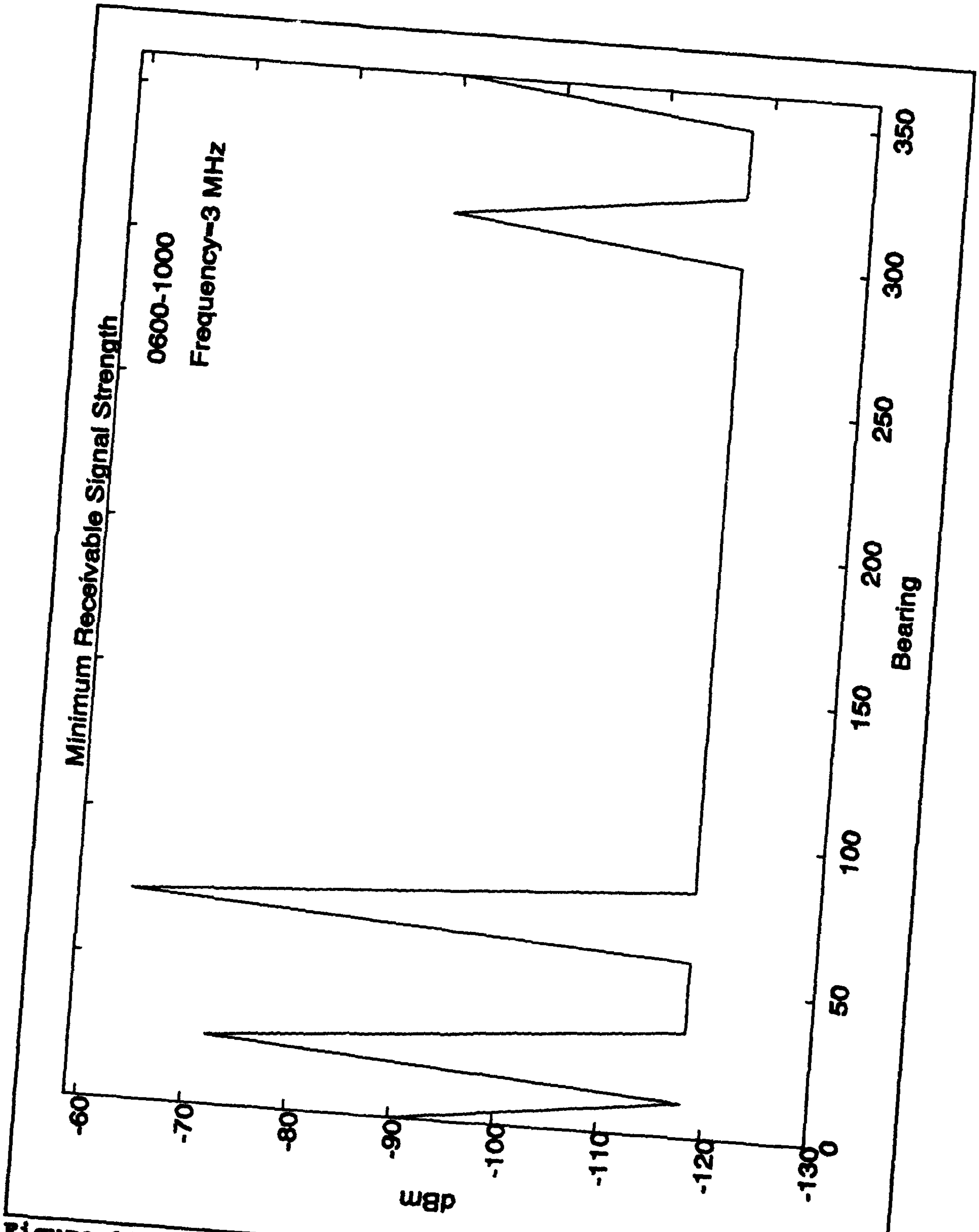

Figure 39 Minimum Signal Strength vs. Bearing

screen would be stored in a file seabag.met. MATLAB does not overwrite data in a .met file, instead it appends data. Therefore, to store in one file all plots resulting from a MAPS session, enter the same filename in single quotes after each query.

When using the latter method, the actual hard copy cannot be obtained until exiting MAPS and MATLAB. The GPP.EXE command associated with MATLAB is then used to produce the hard copy for a specific output device.

IV. PET TECHNIQUE COMPARISON

In this chapter, each of the three PET techniques, manual, semi-automated and MAPS are compared. The important areas of speed, cost, accuracy and ease-of-use are considered.

A. SPEED

In the previous chapter, it was mentioned that more than 20,000 PET plot computations are required to characterize the results of a full PET survey of a typical site. For a performance evaluation system to complete this number of computations, it must be relatively fast. The manual method, using pencil and paper or using drawing software, is by far the slowest. A single PET performance plot, shown in Figure 18, using this method takes approximately ten minutes to complete. The only feasible way to rapidly generate a plot of "percent SOI lost" versus bearing, frequency or time is to precalculate and store each of the 20,000-plus different combinations. Therefore, conducting a full survey using the manual technique is unfeasible.

In this thesis research, two approaches to automation were compared for use with PET, one based on GRAFTOOL and the other on MATLAB software. The semi-automated technique using GRAFTOOL, is somewhat faster than the manual method described above, but also requires a full PET plot, shown in Figure 21,

for each of the combinations of the independent variables. Each plot takes approximately 5 minutes to complete using GRAFTOOL. The GRAFTOOL method allows for the storage of "percent SOI lost" after a PET plot is completed. However, a PET plot must be generated before a plot of "percent SOI lost" versus bearing, frequency or time can be generated.

The MAPS method, based on MATLAB and developed in this thesis, provides a significant advantage over the manual and GRAFTOOL methods. Through MAPS, a single PET plot (or display) can be computed in seconds. Because of the data file structure used by MAPS and the equations developed in this thesis research, MAPS is capable of producing a plot of "percent SOI lost" versus bearing, frequency or time without precomputing and storing a PET plot for each of the 20,000 or so different combinations of bearing, frequency and time. Rather than drawing each of the PET plots required by either the manual method or the GRAFTOOL method, MAPS utilizes the equations and is able to calculate "percent SOI lost" without using construction techniques. Therefore, once system data (noise floor, RFD gain/loss, excess noise) and site data (maximum signal strength, noise interference level) is loaded into MAPS files, a full-blown survey (all variations of the independent variables bearing, frequency, and time are considered) can be completed in minutes.

B. COST

In comparing the three PET techniques, the cost of the computer hardware and software is not a significant factor compared to the operator man-hour costs. Additionally, all three methods can use the same computer system. To be specific, the manual method utilizes a $250 DRAWPERFECT software package. The GRAFTOOL semi-automated method requires a $180 math co-processor and a $200 GRAFTOOL software package. MAPS also requires a math co-processor and a $500 MATLAB software package.

Data collection will not be considered as a cost-trade factor because, currently, all three methods require the same manual procedure. However, if automated data collection is implemented, MAPS can be easily adapted to accept the automated data format.

The total man-hours required to utilize each of the PET models will be broken down into two categories: training time and execution time. The manual method using DRAWPERFECT requires the most training man-hours. Not only does the user have to master PET construction techniques, but also the ambiguities of the software. The GRAFTOOL technique is similar, but not as much training in PET construction is necessary. However, the steps used to create a PET plot in GRAFTOOL must be precisely followed; therefore, a thorough knowledge of the software is required. Because MAPS software is customized for for PET processing, it has been designed to

require minimal training time. A user does not need to know the intricacies involved in PET construction and does not need a complete knowledge of MATLAB. The MAPS menus provide an easy-to-follow system allowing any user to provide PET plots in a very reasonable amount of time. So as far as training costs are concerned, of the three methods, MAPS requires the least amount of training time.

The second category to consider is execution time. In the previous Speed subsection, MAPS was determined to be capable of performing a complete PET survey of a site in minutes. Neither of the other methods could complete a full-blown PET survey in a day. Therefore, as far as execution time is concerned, MAPS is by far the most efficient.

C. ACCURACY

All three methods make use of the linear approximation to the log-normal distribution for SOI. Because of this, all three methods should, in principle, be capable of providing the same level of accuracy. Nevertheless, the GRAFTOOL method and MAPS do provide better accuracy than the manual technique. The improvement in accuracy is primarily due to the fact that computers are capable of more precise calculations than manual curve preparation and scaling.

D. EASE-OF-USE

All three methods require the user to have knowledge of the input parameters as well as the construction of PET plots. The GRAFTOOL method is the most difficult to use. Besides demanding knowledge of PET plots, a very complete knowledge of GRAFTOOL is required. After hours of practice one can become proficient enough to construct plots at a reasonable rate using the GRAFTOOL technique. The manual method is relatively easy to use, but does require a firm grip on construction techniques. The MAPS method is the easiest to use. Unlike the menu-driven GRAFTOOL, MAPS menus are geared toward conducting a full-blown PET analysis of a site. The user does not need an in-depth knowledge of PET plotting techniques, but does require a grasp of the parameters used. However, if the required system input data files (based on measurements) are already in place, anyone can follow MAPS menus and generate the desired performance plots.

E. ADVANTAGES AND DISADVANTAGES

1. MANUAL METHOD

The manual method is the most basic. If using pencil and paper, it is by far the cheapest. However, if graphics software is used, there is no significant cost advantage. The major disadvantage with this method is speed. It is almost impossible to complete a full-blown PET analysis of a site using the manual method.

2. GRAFTOOL SEMI-AUTOMATED Method

The GRAFTOOL method provides a significant increase in speed over the manual method, but it does not provide enough to ensure that a full-blown PET analysis can be completed in a reasonable time period. The primary disadvantage is the complexity of the program. Unless a user has a very firm grasp of GRAFTOOL, difficulties are likely to occur. A major advantage of this technique is that output can be stored in files and used to provide plots such as Percent SOI Lost vs. frequency. Another significant plus is GRAFTOOL's graphical ability. GRAFTOOL is capable of presenting just about any type of graph used in scientific, mathematic and engineering environments. PET plots are simple requirements for such a comprehensive software package.

3. MAPS

The MAPS method provides the speed required to make full-blown PET analyses. Once the required data files have been established, anyone is capable of generating a PET plot or a percent SOI lost vs. frequency or bearing curve. MAPS also is capable of providing percent SOI received without having to type another command or redo an entire plot. The way the data files are established and maintained provides a significant advantage over the other two methods. The site can keep a running file of noise sources. SNEP team members can use the file as a reference when conducting a survey.

Because system parameters, such as system noise floor, RFD gain/loss, and excess noise should not frequently change, it is advantageous to have them stored in files. MAPS permits this. Unlike the manual method or the GRAFTOOL method, MAPS source code is accessible to someone having a knowledge of MATLAB. This permits customization and expedites future improvements.

The major disadvantage occurs with data input. MATLAB only allows the input of .mat files, or data files generated in MATLAB or ASCII files in perfect row-column order to be input. The QBASIC subroutine, CONVER.BAS, was written to alleviate the problem with maximum signal strength data provided by the PROPHET prediction software. Another limitation is that there is, currently, no way to provide a scaled axis on MATLAB 3-D plots. A contour plot is presented with every MAPS 3-D plot to help with plot interpretation. According to Math Works, Inc., the future MATLAB version 4.0 will be capable of providing axes for 3-D plots.

V. CONCLUSIONS AND RECOMMENDATIONS

Skimmons alluded in the Conclusions Section of his thesis that, "...even further benefits can be obtained by writing custom computer software to replace the standard data processing software used..." [Ref. 1]. MAPS is this custom computer software. The manual PET processing method and the GRAFTOOL semi-automated are unsuitable for complete site PET analyses. Neither method, utilizing previously stored data, provides the necessary speed or user-ease to ensure efficient use. On the other hand, MAPS, is an easy-to-use system capable of completing a full-blown PET survey of a site within minutes, assuming the raw data has previously been collected.

Currently, SNEP teams still utilize manual PET processing techniques and manual data collection. MAPS can significantly reduce the data processing time, but cannot reduce any of the data collection time. As discussed, the collection of data through manual means is extremely time-intensive. Since PET processing time has been significantly reduced, the next step is to reduce the time required for data collection. This can be accomplished only through an automated system capable of making accurate RFD loss, excess noise, and noise power measurements.

Many sites now utilize the NEEACT PAC Automatic Noise Measurement System (ANMS) to provide automated noise power

measurements. However, this system is not adequate for adaptation to PET processing and subsequent connection to a computer equipped with MAPS. An advanced ANMS utilizing a new HP FFT Spectrum Analyzer, a high dynamic range A/D translator, band-pass filters, better signal processing software, and a 486-based computer could feasibly produce and store the required data to commence PET analyses. This new data collection/storage system, coupled with MAPS, would allow sites to perform complete PET analysis within minutes. [Ref. 6]

Another important observation deals with the ionospheric propagation prediction programs used to provide the maximum signal strength received by a site. Since this parameter sets the upper limit of the SOI distribution curve, it is imperative that this value be closely approximated to ensure accurate PET results. Current prediction programs usually provide only expected (i.e., mean of a large number of sample observations) values of received signal level. For some applications it would be useful to have additional statistical parameters, such as, the standard deviation or the confidence level of the received signal strength. This could translate into a range of values for the Percent Signals Lost, with an associated confidence. Thus the likelihood that any single SOI observation would exceed the mean detection threshold would be determined. The operational value of such additional

information, as well as modeling and software requirements, could be investigated.

In conclusion, PET analysis is an invaluable tool that site managers need now. The MAPS software, developed during the research reported in this thesis, reduces the processing time so comprehensive PET analyses of sites can be accomplished. However, the major problem of data collection time still restricts up-to-the minute PET analysis. Therefore, it is recommended that research be conducted into the development of a measurement system, similar to that described above, which is capable of automating the measurement of the signal and noise inputs required by MAPS. Once this automated system is in place, site managers could then make use of the wealth of diagnostic information presented through PET analyses.

APPENDIX A. MAPS COMPUTER CODE

MAPS.M

The following is the main computer program used by MAPS to coordinate function calls that provide the selected PET analysis output.

```
c=0;
while c~=9,
clc;
clg;
hold off;
home;
disp('1:  PET Scratch Pad ');
disp('2:  Compute Signals lost based on stored RFD Losses');
disp('3:  Compute Signals lost due to stored Excess Noise');
disp('4:  Compute Signals lost due to stored Noise
      Sources');
disp('5:  Signal Reception Determination');
disp('6:  System Parameter Files Generation/Manipulation');
disp('7:  Noise Source Files Generation/Manipulation');
disp('8:  Initialize/Edit Maximum Signal Strength Files');
disp('9:  Exit MAPS');
c=input('Command #? ');
brng=[0:12:360];
freq=[2:32];
a1=[0 359 0 100];
a2=[2 32 0 100];
a3=[2 32 0 360];
if c==1,
   mss=input('Maximum Signal Strength (dBm)? ');
   nf=input('Noise Floor (dBm)? ');
   rfd=input('RFD Loss/Gain (dB)? ');
   np=input('Noise Source Strength (dBm) (Enter -999 if
             none)?     ');
   l=loss(nf,mss,rfd,np);
   a=avail(nf,mss,rfd,np);
   plotpet(l,nf,mss,rfd,np);
   text(.65,.20,sprintf('Pct Lost=%g',l),'sc')
   text(.65,.15,sprintf('Pct Rcvd=%g',a),'sc')
   pause;
   outfile;
elseif c==2,
   disp('Compute RFD Losses for which system
         (rfdsystem.mat):');
   dir rfd*.mat
   sys=input('Enter System Name (in Single Quotes)? ');
```

```
tim=intime;
mss=loadmss(tim);
rfd=loadrfdf(sys);
nf=loadnff(sys);
o=optout;
if o==1,
   f=infreq;
   b=inbrng;
   l=loss(nf,mss(b,f),rfd(b,f),-999);
   a=avail(nf,mss(b,f),rfd(b,f),-999);
   plotpet(l,nf,mss(b,f),rfd(b,f),-999)
   text(.65,.20,sprintf('Pct Lost=%g',l),'sc');
   text(.65,.15,sprintf('Pct Rcvd=%g',a),'sc');
   text(.15,.90,outtime(tim),'sc');
   text(.15,.85,sprintf('Frequency=%g Mhz',f+1),'sc');
   text(.15,.80,sprintf('Bearing=%g',(b-1)*12),'sc');
   pause;
   outfile;
elseif o==2,
   f=infreq;
   l=loss(nf,mss(:,f),rfd(:,f),-999);
   a=avail(nf,mss(:,f),rfd(:,f),-999);
   axis(a1);
   subplot(211);
   plot(brng,l);grid;title('Signals Lost due to RFD');
   xlabel('Bearing');ylabel('% Signals Lost');
   text(6,89,outtime(tim));
   text(6,81,sprintf('Frequency=%g Mhz',f+1));
   subplot(212);
   plot(brng,a);grid;title('Signals Received');
   xlabel('Bearing');ylabel('% Signals Received');
   text(6,12,outtime(tim));
   text(6,4,sprintf('Frequency=%g Mhz',f+1));
   pause;
   outfile;
elseif o==3,
   b=inbrng;
   l=loss(nf,mss(b,:),rfd(b,:),-999);
   a=avail(nf,mss(b,:),rfd(b,:),-999);
   axis(a2);
   subplot(211);
   plot(freq,l);grid;title('Signals Lost due to RFD');
   xlabel('Frequency (Mhz)');ylabel('% Signals Lost');
   text(3,89,outtime(tim));
   text(3,81,sprintf('Bearing=%g',(b-1)*12));
   subplot(212);
   plot(freq,a);grid;title('Signals Received');
   xlabel('Frequency (Mhz)');ylabel('% Signals
         Received');
   text(3,12,outtime(tim));
   text(3,4,sprintf('Bearing=%g',(b-1)*12));
```

```
        pause;
        outfile;
     elseif o==4,
        l=loss(nf,mss,rfd,-999);
        subplot(211);
        mesh(l);title('3D Plot of Signals Lost due to RFD');
        text(.45,.9,outtime(tim),'sc');
        subplot(212);
        nl=revrow(l);
        axis(a3);
        contour(nl,10,freq,brng);
        xlabel('Frequency (Mhz)');ylabel('Bearing');
        title('Contour Plot of Signals Lost due to RFD');
        pause;
        outfile;
     end;
  elseif c==3,
     disp('Compute Losses due to Excess Noise for which system
           (ensystem.mat):');
     dir en*.mat
     sys=input('Enter System Name (in Single Quotes)? ');
     tim=intime;
     mss=loadmss(tim);
     en=loadenf(sys);
     rfd=loadrfdf(sys);
     nf=loadnff(sys);
     o=optout;
     if o==1,
        f=infreq;
        b=inbrng;
        l=loss(nf,mss(b,f),rfd(b,f),nf+en(b,f));
        a=avail(nf,mss(b,f),rfd(b,f),nf+en(b,f));
        plotpet(l,nf,mss(b,f),rfd(b,f),nf+en(b,f));
        text(.65,.20,sprintf('Pct Lost=%g',l),'sc');
        text(.65,.15,sprintf('Pct Rcvd=%g',a),'sc');
        text(.15,.90,outtime(tim),'sc');
        text(.15,.85,sprintf('Frequency=%g Mhz',f+1),'sc');
        text(.15,.80,sprintf('Bearing=%g',(b-1)*12),'sc');
        pause;
        outfile;
     elseif o==2,
        f=infreq;
        l=lossn(nf,mss(:,f),rfd(:,f),nf+en(:,f));
        a=availn(nf,mss(:,f),rfd(:,f),nf+en(:,f));
        axis(a1);
        subplot(211);
        plot(brng,l);grid;title('Signals Lost due to Excess
                                 Noise');
        xlabel('Bearing');ylabel('% Signals Lost');
        text(6,89,outtime(tim));
        text(6,81,sprintf('Frequency=%g Mhz',f+1));
```

```
      subplot(212);
      plot(brng,a);grid;title('Signals Received');
      xlabel('Bearing');ylabel('% Signals Received');
      text(6,12,outtime(tim));
      text(6,4,sprintf('Frequency=%g Mhz',f+1));
      pause;
      outfile;
   elseif o==3,
      b=inbrng;
      l=lossn(nf,mss(b,:),rfd(b,:),nf+en(b,:));
      a=availn(nf,mss(b,:),rfd(b,:),nf+en(b,:));
      axis(a2);
      subplot(211);
      plot(freq,l);grid;title('Signals Lost due to Excess
                                Noise');
      xlabel('Frequency (Mhz)');ylabel('% Signals Lost');
      text(3,89,outtime(tim));
      text(3,81,sprintf('Bearing=%g',(b-1)*12));
      subplot(212);
      plot(freq,a);grid;title('Signals Received');
      xlabel('Frequency (Mhz)');ylabel('% Signals
                                         Received');
      text(3,12,outtime(tim));
      text(3,4,sprintf('Bearing=%g',(b-1)*12));
      pause;
      outfile;
   elseif o==4,
      l=lossn(nf,mss,rfd,nf+en);
      subplot(211);
      mesh(l);title('3D Plot of Signals Lost due to Excess
                    Noise');
      text(.45,.9,outtime(tim),'sc');
      subplot(212);
      nl=revrow(l);
      axis(a3);
      contour(nl,10,freq,brng);
      xlabel('Frequency (Mhz)');ylabel('Bearing');
      title('Contour Plot of Signals Lost due to Excess
             Noise');
      pause;
      outfile;
   end;
elseif c==4,
   disp('Compute Losses due to Noise sources for which
         system (rfdsystem.mat):');
   dir rfd*.mat
   sys=input('Enter System Name (in single quotes)? ');
   tim=intime;
   mss=loadmss(tim);
   rfd=loadrfdf(sys);
   en=loadenf(sys);
```

```
nf=loadnff(sys);
np=loadnoisef(tim);
for i=1:31
    for j=1:31
      if nf+en(i,j)>np(i,j)
         np(i,j)=nf+en(i,j);
      end;
    end;
 end;
o=optout;
if o==1,
   f=infreq;
   b=inbrng;
   l=loss(nf,mss(b,f),rfd(b,f),np(b,f));
   a=avail(nf,mss(b,f),rfd(b,f),np(b,f));
   plotpet(l,nf,mss(b,f),rfd(b,f),np(b,f));
   text(.65,.20,sprintf('Pct Lost=%g',l),'sc');
   text(.65,.15,sprintf('Pct Rcvd=%g',a),'sc');
   text(.15,.90,outtime(tim),'sc');
   text(.15,.85,sprintf('Frequency=%g Mhz',f+1),'sc');
   text(.15,.80,sprintf('Bearing=%g',(b-1)*12),'sc');
   pause;
   outfile;
elseif o==2,
   f=infreq;
   l=lossn(nf,mss(:,f),rfd(:,f),np(:,f));
   a=availn(nf,mss(:,f),rfd(:,f),np(:,f));
   axis(a1);
   subplot(211);
   plot(brng,l);grid;title('Signals Lost due to Noise');
   xlabel('Bearing');ylabel('% Signals Lost');
   text(6,89,outtime(tim));
   text(6,81,sprintf('Frequency=%g Mhz',f+1));
   subplot(212);
   plot(brng,a);grid;title('Signals Received');
   xlabel('Bearing');ylabel('%Signals Received');
   text(6,12,outtime(tim));
   text(6,4,sprintf('Frequency=%g Mhz',f+1));
   pause;
   outfile;
elseif o==3,
   b=inbrng;
   l=lossn(nf,mss(b,:),rfd(b,:),np(b,:));
   a=availn(nf,mss(b,:),rfd(b,:),np(b,:));
   axis(a2);
   subplot(211);
   plot(freq,l);grid;title('Signals Lost due to Noise');
   xlabel('Frequency (Mhz)');ylabel('% Signals Lost');
   text(3,89,outtime(tim));
   text(3,81,sprintf('Bearing=%g',(b-1)*12));
   subplot(212);
```

```
        plot(freq,a);grid;title('Signals Received');
        xlabel('Frequency (MHz)');ylabel('% Signals
                                        Received');
        text(3,12,outtime(tim));
        text(3,4,sprintf('Bearing=%g',(b-1)*12));
        pause;
        outfile;
     elseif o==4,
        l=lossn(nf,mss,rfd,np);
        subplot(211);
        mesh(l);title('3D Plot of Signals Lost due to Noise');
        text(.45,.9,outtime(tim),'sc');
        subplot(212);
        nl=revrow(l);
        axis(a3);
        contour(nl,10,freq,brng);
        xlabel('Frequency (Mhz)');ylabel('Bearing');
        title('Contour Plot of Signals Lost due to Noise');
        pause;
        outfile;
     end;
  elseif c==5,
     disp('Compute Minimum Receivable Signal Strengths for ');
     disp('which system (rfdsystem.mat):');
     dir rfd*.mat
     sys=input('Enter System Name (in single quotes)? ');
     tim=intime;
     rfd=loadrfdf(sys);
     en=loadenf(sys);
     nf=loadnff(sys);
     np=loadnoisef(tim);
     minss=minsig(nf,rfd,en,np);
     mn=nf-5;
     clc;
     disp('1:  Frequency and Bearing Known');
     disp('2:  Frequency Known');
     disp('3:  Bearing Known');
     sc=input('Command? ');
     if sc==1,
        f=infreq;
        b=inbrng;
        disp(sprintf('Frequency = %g MHz',f+1));
        disp(sprintf('Bearing = %g',(b-1)*12));
        disp(sprintf('Minimum receivable signal = %g
                      dBm',minss(b,f)));
     elseif sc==2,
        f=infreq;
        mx=max(minss(:,f))+5;
        a4=[0 360 mn mx];
        axis(a4);
        plot(brng,minss(:,f));title('Minimum Receivable Signal
```

```
                                             Strength');
      xlabel('Bearing');ylabel('dBm');
      text(.7,.9,outtime(tim),'sc');
      text(.7,.85,sprintf('Frequency=%g MHz',f+1),'sc');
      pause;
      outfile;
   elseif sc==3,
      b=inbrng;
      mx=max(minss(b,:))+5;
      a4=[0 32 mn mx];
      axis(a4);
      plot(freq,minss(b,:));title('Minimum Receivable Signal
                                             Strength');
      xlabel('Frequency (MHz)');ylabel('dBm');
      text(.75,.9,outtime(tim),'sc');
      text(.75,.85,sprintf('Bearing=%g',(b-1)*12),'sc');
      pause;
      outfile;
   end;
elseif c==6,
   clc;
   disp('1:  Setup a system');
   disp('2:  View/Edit System RFD Gain/Loss File');
   disp('3:  View/Edit System Excess Noise File');
   disp('4:  View/Edit System Noise Floor');
   cmd=input('Command? ');
   if cmd==1,
      sys=input('System to be setup, in single quotes? ');
      setnf(sys);
      matrfd(sys);
      maten(sys);
   elseif cmd==2,
      dir rfd*.mat
      sys=input('Enter system to edit, in single quotes
                  (rfdsystem.mat)? ');
      edrfd(sys);
   elseif cmd==3,
      dir en*.mat
      sys=input('Enter system to edit, in single quotes
                  (ensystem.mat)? ');
      eden(sys);
   elseif cmd==4,
      dir nf*.mat
      sys=input('Enter system to edit, in single quotes
                  (nfsystem.mat)? ');
      ednf(sys);
   end;
elseif c==7,
   clc;
   disp('1:  Initialize Noise Source Files');
   disp('2:  Edit/View Noise Source Files');
```

```
   cmd=input('Enter Command? ');
   if cmd==1,
      initnoise;
   elseif cmd==2,
      ednoise;
   end;
elseif c==8,
   clc;
   disp('1: Initialize new maximum signal strength files');
   disp('2: Edit current maximum signal strength files');
   disp('3: View current maximum signal strength files');
   cmd=input('Enter Command? ');
   if cmd==1,
      initmss;
   elseif cmd==2,
       edmss;
   elseif cmd==3,
       tim=intime;
       mss=loadmss(tim);
       mn=-207;
       mx=max(max(mss))+1;
       a5=[2 32 mn mx];
       a6=[0 360 mn mx];
       subplot(211);
       axis(a5);
       plot(freq,mss);title('Maximum Signal Strength');
       xlabel('Frequency (MHz)');ylabel('MSS (dBm)');
       text(.75,.9,outtime(tim),'sc');
       subplot(212);
       axis(a6);
       plot(brng,mss');title('Maximum Signal Strength');
       xlabel('Bearing');ylabel('MSS (dBm)');
       text(.75,.9,outtime(tim),'sc');
       pause;
       outfile;
    end;
elseif c==9,
else disp('Improper Command');
end;
end;
```

AVAIL.M

This function uses the input paramaters (vectors): system noise floor, maximum signal strength, RFD gain/loss, and noise signal strength, and determines the Percent Signals Received.

```
function [a]=avail(nf,mss,rfd,np);
[r,c]=size(rfd);
```

```
for i=1:r
   for j=1:c
if rfd(i,j)<0,
   rfd(i,j)=abs(rfd(i,j));
else rfd(i,j)=0;
end;
end;
end;
m=100 ./(nf-mss);
if np<=nf,
     a=((nf+12+rfd).*m)-(m.*mss);
else a=((np+12+rfd).*m)-(m.*mss);
end;
for i=1:r
  for j=1:c
    if mss(i,j)<=nf,
       a(i,j)=0;
    end;
    if a(i,j)>100,
       a(i,j)=100;
    elseif a(i,j)<0,
       a(i,j)=0;
    end;
  end;
end;
a=round(10*a)/10;
```

AVAILN.M

This function is identical to AVAIL.M; however, this function returns Percent Signals Received for matrices.

```
function [a]=availn(nf,mss,rfd,np);
[r,c]=size(rfd);
[rn,cn]=size(np);
for i=1:r
   for j=1:c
if rfd(i,j)<0,
   rfd(i,j)=abs(rfd(i,j));
else rfd(i,j)=0;
end;
end;
end;
m=100 ./(nf-mss);
for i=1:rn
   for j=1:cn
       if mss(i,j)>nf,
          if np(i,j)<=nf,

a(i,j)=((nf+12+rfd(i,j))*m(i,j))-(m(i,j)*mss(i,j));
```

```
          else
a(i,j)=((np(i,j)+12+rfd(i,j))*m(i,j))-(m(i,j)*mss(i,j));
          end;
          if a(i,j)>100,
             a(i,j)=100;
          elseif a(i,j)<0,
             a(i,j)=0;
          end;
        else a(i,j)=0;
        end;
    end;
end;
a=round(10*a)/10;
```

EDNF.M

This function displays a system's noise floor and allows for it to be changed.

```
function []=ednf(sys)
r=' nf';
eval(['load',r,sys]);
disp(sprintf('Current Noise Floor = %g dbm',nf));
y=input('Enter 1 to enter new Noise Floor, 0 to continue
         without change? ');
if y==1,
   nf=input('New Noise Floor (dBm)? ');
   eval(['save',r,sys]);
end;
```

EDEN.M

This function allows a system's excess noise file to be edited or viewed.

```
function []=eden(sys)
clc;
disp('1:  Replace Excess Noise Values');
disp('2:  View Excess Noise File');
c=input('Desired Command? ');
en=loadenf(sys);
if c==1,
   flag=0;
   while flag~=1,
   f=infreq;
   b=inbrng;
   p=input('New Excess Noise Level in dB? ');
   disp(sprintf('An Excess Noise of %g dB is already
        stored.',en(b,f)));
```

```
   disp('Enter 1, Overwrite with new information, or 0, Keep
        Stored Info');
   x=input('? ');
   if x==1
      en(b,f)=p;
      if b==1,
         en(31,f)=p;
      elseif b==31,
         en(1,f)=p;
      end;
   end;
   flag=input('Enter 1 to Quit, 0 to Continue Editing? ');
   end;
   r=' en';
   eval(['save',r,sys]);
elseif c==2,
   mn=min(min(en))-1;
   mx=max(max(en))+1;
   a1=[2 32 mn mx];
   a2=[0 360 mn mx];
   subplot(211);
   axis(a1);
   plot(2:32,en');title('Excess Noise by Frequency');
   xlabel('Frequency');ylabel('Excess Noise (dB)');
   grid;
   subplot(212);
   axis(a2);
   plot(0:12:360,en);title('Excess Noise by Bearing');
   xlabel('Bearing');ylabel('Excess Noise (dB)');
   grid;
   pause;
   outfile;
end;
```

EDNOISE.M

This function allows a site's noise files to be viewed or edited.

```
clc;
disp('1:  Add new Noise Source');
disp('2:  Delete a Noise Source');
disp('3:  View Noise File');
c=input('Desired Command? ');
t=intime;
np=loadnoise(t);
if c==1,
   flag=0;
   while flag~=1,
   f=infreq;
```

```
   b=inbrng;
   p=input('Noise Source Strength (dBm)? ');
   if np(b,f)==-999
      np(b,f)=p;
      if b==1,
         np(31,f)=p;
      elseif b==31,
         np(1,f)=p;
      end;
   else,
      disp(sprintf('A noise source of %g dBm is already
                    stored.',np(b,f)));
      disp('Enter 1, Overwrite with new information, or 0,
            Keep Stored Info');
      x=input('? ');
      if x==1
         np(b,f)=p;
         if b==1,
            np(31,f)=p;
         elseif b==31,
            np(1,f)=p;
         end;
      end;
   end;
      flag=input('Enter 1 to Quit, 0 to Continue Adding
                  Noise Sources? ');
   end;
   savenoise(t,np);
elseif c==2,
   flag=0;
   while flag~=1,
   f=infreq;
   b=inbrng;
   disp('Delete Noise source')
   disp([sprintf('Frequency %g Mhz',f+1),sprintf('   Bearing
                           %g',(b-1)*12)]);
   x=input('Enter 1 to delete signal? ');
   if x==1,
      np(b,f)=-999;
      if b==1,
         np(31,f)=-999;
      elseif b==31,
         np(1,f)=-999;
      end;
   end;
   flag=input('Enter 1 to quit, or 0 to continue deleting
               Noise');
   end;
   savenoise(t,np);
elseif c==3,
   for i=1:31
```

```
      for j=1:31
        if np(i,j)==-999,
           np(i,j)=-150;
        end;
      end;
    end;
    mn=-155;
    mx=max(max(np))+1;
    a1=[2 32 mn mx];
    a2=[0 360 mn mx];
    subplot(211);
    axis(a1);
    plot(2:32,np');title('Noise Sources by Frequency');
    xlabel('Frequency');ylabel('Noise Power (dBm)');
    text(.30,.91,'-150 dBm represents No Noise Sources
                  Present','sc');
    grid;
    subplot(212);
    axis(a2);
    plot(0:12:360,np);title('Noise Sources by Bearing');
    xlabel('Bearing');ylabel('Noise Power (dBm)');
    text(.30,.41,'-150 dBm represents No Noise Sources
                  Present','sc');
    grid;
    pause;
    outfile;
end;
```

EDMSS.M

This function allows converted PROPHET data files to be added to the current maximum signal strength files. This is important for PROPHET files with different bearings.

```
function []=edmss
disp('Input converted PROPHET data filename, do not include
    .DAT extension');
f=input('Name of file to add to MAPS mss files (in single
         quotes)? ');
eval(['load ',f,'.dat']);
eval(['load ',f,'1.dat']);
m=eval(f);
m1=eval([f,'1']);
j=1;
disp('Input Signal Bearing');
b=inbrng;
for k=1:6
    mss=loadmss(k);
    for i=1:19
       mss(b,i)=max(m(j:j+3,i))-107;
```

```
      end;
      l=20;
      for i=1:6
          mss(b,l)=max(ml(j:j+3,i))-107;
          mss(b,l+1)=mss(b,l);
          l=l+2;
      end;
if b==1,
   mss(31,:)=mss(1,:);
elseif b==31,
   mss(1,:)=mss(31,:);
end;
savemss(mss,k);
j=j+4;
end;
```

EDRFD.M

This function allows for system RFD gain/loss files to be viewed or edited.

```
function []=edrfd(sys)
clc;
disp('1:  Replace RFD Gain(+)/Loss(-) Values');
disp('2:  View RFD Gain/Loss File');
c=input('Desired Command? ');
rfd=loadrfdf(sys);
if c==1,
   flag=0;
   while flag~=1,
   f=infreq;
   b=inbrng;
   p=input('New RFD Gain(+)/Loss(-) in dB? ');
   disp(sprintf('A RFD Gain/Loss of %g dB is already
                 stored.',rfd(b,f)));
   disp('Enter 1, Overwrite with new information, or 0, Keep
         Stored Info');
   x=input('? ');
   if x==1
      rfd(b,f)=p;
      if b==1,
         rfd(31,f)=p;
      elseif b==31,
         rfd(1,f)=p;
      end;
   end;
    flag=input('Enter 1 to Quit, 0 to Continue Editing? ');
   end;
   r=' rfd';
   eval(['save',r,sys]);
```

```
elseif c==2,
   subplot(211);
   plot(2:32,rfd);title('RFD Gain/Loss by Frequency');
   xlabel('Frequency');ylabel('Gain/Loss (dB)');
   subplot(212);
   plot(0:12:359,rfd');title('RFD Gain/Loss by Bearing');
   xlabel('Bearing');ylabel('Gain/Loss (dB)');
   pause;
   outfile;
end;
```

INBRNG.M

This function calls for the input of a bearing.

```
function [b]=inbrng
b=round(input('Bearing? ')/12)+1;
if b>31,
   b=b-30;
end;
```

INFREQ.M

This function calls for the input of a frequency in MHz.

```
function [f]=infreq
clc;
f=round(input('Frequency (2-32 Mhz)? '))-1;
```

INITMSS.M

This function initializes the 6 maximum strength files. It converts a single converted PROPHET data file into the frequency and bearing dependent matrices used by MAPS.

```
function []=initmss
disp('Input converted PROPHET data filename, do not include
      .DAT extension');
f=input('Name of file to be initialized (in single quotes)?
       ');
eval(['load ',f,'.dat']);
eval(['load ',f,'1.dat']);
m=eval(f);
m1=eval([f,'1']);
j=1;
for k=1:6
    for i=1:19
       mss(1,i)=max(m(j:j+3,i));
```

```
     end;
     l=20;
     for i=1:6
         mss(1,l)=max(m1(j:j+3,i));
         mss(1,l+1)=mss(1,l);
         l=l+2;
     end;
     for i=2:31
          mss(i,:)=mss(1,:);
     end;
mss=mss-107;
savemss(mss,k);
j=j+4;
end;
```

INITNOISE.M

This function intializes the 6 noise source files. All files are set to the default value of -999 dBm.

```
disp('This function initializes the six Noise Source
      Files');
disp('It will erase, the current noise files');
disp('Type ctrl-c to quit, or return to continue');
pause;
for i=1:31
   for j=1:31
     np(i,j)=-999;
   end;
end;
save noise1 np
save noise2 np
save noise3 np
save noise4 np
save noise5 np
save noise6 np
end
```

INTIME.M

This function calls for one of the specified time periods to be selected.

```
function [t]=intime
clc;
disp('All Times are Zulu');
disp('1:  0600 - 1000');
disp('2:  1000 - 1400');
disp('3:  1400 - 1800');
disp('4:  1800 - 2200');
```

```
disp('5:  2200 - 0200');
disp('6:  0200 - 0600');
t=input('Choose desired time period? ');
```

LOADENF.M

This function loads the selected system's excess noise file.

```
function [en]= loadenf(sys)
r='en';
eval(['load ',r,sys]);
end;
```

LOADMSS.M

This funciton loads the maximum signal strength files of a specified time period.

```
function [mss]=loadmss(t)
if t==1,
   load mss1
elseif t==2,
   load mss2
elseif t==3,
   load mss3
elseif t==4,
   load mss4
elseif t==5,
   load mss5
elseif t==6,
   load mss6
end;
```

LOADNFF.M

This function loads a selected system's noise floor level.

```
function [nf]=loadnff(sys)
r=' nf';
eval(['load',r,sys]);
```

LOADNOISEF.M

This function loads the noise source file corresponding to the selected time period.

```
function [np]=loadnoise(w);
if w==1,
   load noise1;
elseif w==2,
   load noise2;
elseif w==3,
   load noise3;
elseif w==4,
   load noise4;
elseif w==5,
   load noise5;
elseif w==6,
   load noise6;
end;
```

LOADRFDF.M

This function loads a selected system's RFD gain/loss file.

```
function [rfd] = loadrfdf(sys)
r='rfd';
eval(['load ',r,sys]);
end;
```

LOSS.M

This function computes the Percent Signals Lost due to system noise floor, RFD gain/loss, and noise power. This function works only on parameters that are vectors.

```
function [l]=loss(nf,mss,rfd,np);
[r,c]=size(rfd);
[rn,cn]=size(np);
for i=1:r
   for j=1:c
if rfd(i,j)<0,
   rfd(i,j)=abs(rfd(i,j));
else rfd(i,j)=0;
end;
end;
end;
m=100 ./(mss-12-nf);
if np<=nf,
     l=m.*rfd;
else l=m.*(np+rfd-nf);
end;
for i=1:r
  for j=1:c
```

```
    if mss(i,j)<=nf
       l(i,j)=0;
    end;
    if l(i,j)>100,
       l(i,j)=100;
    elseif l(i,j)<0,
       l(i,j)=0;
    end;
  end;
end;
l=round(10*l)/10;
```

LOSSN.M

This function computes the Percent Signals Lost due to system noise floor, RFD gain/loss, and noise power. This function works only on parameters that are matrices.

```
function [l]=lossn(nf,mss,rfd,np);
[r,c]=size(rfd);
[rn,cn]=size(np);
for i=1:r
   for j=1:c
if rfd(i,j)<0,
   rfd(i,j)=abs(rfd(i,j));
else rfd(i,j)=0;
end;
end;
end;
m=100 ./(mss-12-nf);
for i=1:rn
   for j=1:cn
       if mss(i,j)>nf,
          if np(i,j)<=nf,
             l(i,j)=m(i,j)*rfd(i,j);
          else l(i,j)=m(i,j)*(np(i,j)+rfd(i,j)-nf);
          end;
          if l(i,j)>100,
             l(i,j)=100;
          elseif l(i,j)<0,
             l(i,j)=0;
          end;
       else l(i,j)=0;
       end;
   end;
end;
l=round(10*l)/10;
```

MATEN.M

This function creates a system's excess noise file.

```
function []=maten(sys)
en=zeros(31,31);
for i=1:31
  disp(sprintf('%g Mhz',i+1));
  en(1,i)=input('Excess Noise (dB)? ');
end;
for j=1:31
  en(j,:)=en(1,:);
end;
r=' en';
eval(['save ',r,sys]);
end;
```

MATRFD.M

This function creates a system's RFD gain/loss file.

```
function []=matrfd(sys)
rfd=zeros(31,31);
for i=1:31
  disp(sprintf('%g Mhz',i+1));
  rfd(1,i)=input('RFD Gain(+)/Loss(-)? ');
end;
for j=1:31
  rfd(j,:)=rfd(1,:);
end;
r=' rfd';
eval(['save ',r,sys]);
end;
```

MINSIG.M

This function computes the minimum amount of signal power required to receive it. This function does not take into consideration a system's threshold.

```
function [ms]=minsig(nf,rfd,en,np)
for i=1:31
  for j=1:31
      if rfd(i,j)<=0,
         ms(i,j)=nf-rfd(i,j);
      else ms(i,j)=nf;
      end;
      if nf+en(i,j)>=ms(i,j),
         ms(i,j)=nf+en(i,j);
```

```
      end;
      if np(i,j)>=ms(i,j),
         ms(i,j)=np(i,j);
      end;
   end;
end;
```

OPTOUT.M

This function calls for the selection of one of the four output choices.

```
function [o]=optout
clc;
disp('Choose desired output option');
disp('1:  1 Frequency / 1 Bearing');
disp('2:  1 Frequency / All Bearings');
disp('3:  All Frequencies / 1 Bearing');
disp('4:  All Frequencies / All Bearings (3D Plot)');
o=input('Desired Option? ');
```

OUTFILE.M

This function controls the storage of MAPS results.

```
function []=outfile
clc;
disp('1:  Save plot to .met file');
disp('2:  Do not save and continue');
i=input('Enter 1 or 2? ');
if i==1,
   disp('Same filename can be used, plots will be
appended.');
   n=input('Name of plot file (in single quotes)? ');
   eval(['meta ',n]);
end;
```

OUTTIME.M

This function allows for the display of selected time periods.

```
function [tim]=outtime(t)
if t==1,
   tim='0600-1000';
elseif t==2,
   tim='1000-1400';
elseif t==3,
```

```
   tim='1400-1800';
elseif t==4,
   tim='1800-2200';
elseif t==5,
   tim='2200-0200';
elseif t==6,
   tim='0200-0600';
end;
```

PLOTINT.M

This function plots the intersection that was used to compute Percent Signals Lost.

```
function []=plotint(start,h,v)
for i=start:h
    y(i-start+1)=v;
end;
for j=0:v
    x(j+1)=h;
end;
plot(start:h,y,x,0:v)
end;
```

PLOTPET.M

This function provides the basic PET plot.

```
function [ ] = plotpet(l,nf,mss,rfd,np);
if rfd<0,
   rfd=abs(rfd);
else rfd=0;
end;
y=[0:100];
m=100/(mss-12-nf);
plot(y./m+nf+12,y,y./m+nf,y,y./m+nf-rfd,y);
a=axis;
hold on;
if np<=nf,
   plotint(a(1),nf,l);title('Loss of Signals due to RFD');
   xlabel('dBm');ylabel('% Signals Lost');
else plotint(a(1),np,l);
     title('Loss of Signals due to Noise');
     xlabel('dBm');ylabel('% Signals Lost');
end;
hold off;
```

REVROW.M

This function reverses the rows of a matrix (row 1 to row n).

```
function [nl]=revrow(y)
for i=1:31,
   nl(i,:)=y(31-i,:);
end;
```

SAVENOISE.M

This function saves the desired noise source file.

```
function []=savenoise(t,np)
if t==1,
   save noise1 np
elseif t==2,
   save noise2 np
elseif t==3,
   save noise3 np
elseif t==4,
   save noise4 np
elseif t==5,
   save noise5 np
elseif t==6,
   save noise6 np
end;
```

SETNF.M

This function set's a selected system's noise floor level.

```
function []=setnf(sys)
r=' nf';
nf=input('System Noise Floor (dBm)? ');
eval(['save',r,sys]);
```

CONVERT.BAS

This is the BASIC program written to convert PROPHET data into data useable by MATLAB. The function INITMSS.M and EDMSS.M are set up to use data converted by this program.

```
DIM G$(10), A$(24), A1$(24)
INPUT "NAME OF PROPHET DATA FILE (2-20 MHz) TO BE
```

```
CONVERTED"; F$
INPUT "NAME OF PROPHET DATA FILE (20-32 MHz) TO BE
CONVERTED"; F1$
OPEN F$ FOR INPUT AS #1
OPEN F1$ FOR INPUT AS #2
FOR I = 1 TO 10
    LINE INPUT #1, G$(I)
    LINE INPUT #2, G1$(I)
NEXT I
FOR J = 1 TO 24
    T$ = INPUT$(7, #1)
    T1$ = INPUT$(31, #2)
    PRINT T1$
    LINE INPUT #1, N$
    LINE INPUT #2, N1$
    FOR K = 1 TO 57 STEP 3
        D$ = MID$(N$, K, 3)
        IF D$ = "   " THEN D$ = " -99"
        IF LEFT$(D$, 1) = "-" THEN D$ = " " + D$
        A$(J) = A$(J) + D$
    NEXT K
    FOR K = 1 TO 18 STEP 3
        D1$ = MID$(N1$, K, 3)
        IF D1$ = "   " THEN D1$ = " -99"
        IF LEFT$(D1$, 1) = "-" THEN D1$ = " " + D1$
        A1$(J) = A1$(J) + D1$
    NEXT K
NEXT J
CLOSE #1
CLOSE #2
INPUT "NAME OF CONVERTED FILE"; M$
OPEN M$ + ".DAT" FOR OUTPUT AS #1
OPEN M$ + "1" + ".DAT" FOR OUTPUT AS #2
FOR I = 7 TO 24
    PRINT #1, A$(I)
    PRINT #2, A1$(I)
NEXT I
FOR I = 1 TO 6
    PRINT #1, A$(I)
    PRINT #2, A1$(I)
NEXT I
CLOSE #1
CLOSE #2
END
```

LIST OF REFERENCES

1. Skimmons, Brian E., Automated Performance Evaluation Technique, Master's Thesis, Naval Postgraduate School, Monterey, California, March 1992.

2. Lott, Gus K. Jr., High Frequency (HF) Radio Signal Amplitude Characteristics, HF Receiver Site Performance Criteria, and Expanding the Dynamic Range of HF Digital New Energy Receivers by Strong Signal Elimination, Doctoral Dissertation, Naval Postgraduate School, Monterey, California, June 1990.

3. Gagliardi, Robert M., Introduction to Communications Engineering, Second Edition, pp. 141-163, John Wiley & Sons, Inc., 1988.

4. Delfin Systems DN-91-214, Operators Manual for the Advanced PROPHET System, IWG Corporation, San Diego, California, 1991.

5. Vincent, Wilbur Ray, Adler, Richard W., Wadsworth, Donald V.Z., and others, "Quick-Look Report on SNEP Team Visit to NSGA Hanza", Signal-to-Noise Enhancement Program, April 1992.

6. Vincent, Wilbur Ray, Note to Richard W. Adler and Donald V.Z. Wadsworth, Subject: Future Directions of PET, 26 June 1992.

INITIAL DISTRIBUTION LIST

1.	Defense Technical Information Center Cameron Station Alexandria, VA 22304-6145	2
2.	Library, Code 52 Naval Postgraduate School Monterey, CA 93943-5002	2
3.	Chairman, Code EC Department of Electrical and Computer Engineering Naval Postgraduate School Monterey, CA 93943-5000	1
4.	Chairman, Code EW Electronic Warfare Academic Group Naval Postgraduate School Monterey, CA 93943-5000	1
5.	Professor D. V. Wadsworth, Code EC/Wd Department of Electrical and Computer Engineering Naval Postgraduate School Monterey, CA 93943-5000	2
6.	Professor R. W. Adler, Code EC/Ab Department of Electrical and Computer Engineering Naval Postgraduate School Monterey, CA 93943-5000	5
7.	Professor W. R. Vincent, Code EC/Ab Department of Electrical and Computer Engineering Naval Postgraduate School Monterey, CA 93943-5000	3
8.	Commander Naval Security Group Command Attn: G43 3801 Nebraska Avenue, N.W. Washington, DC 20390	1
9.	Commander Naval Security Group Command Attn: G433 3801 Nebraska Avenue, N.W. Washington, DC 20390	3

10. Commander 1
Naval Security Group Command
Attn: G80
3801 Nebraska Avenue, N.W.
Washington, DC 20390

11. Commander 1
Naval Security Group Command
Attn: G81
3801 Nebraska Avenue, N.W.
Washington, DC 20390

12. Commander 1
Naval Security Group Command
Attn: GX, CDR G. K. Lott
3801 Nebraska Avenue, N.W.
Washington, DC 20390

13. Commander 3
Naval Electronic Systems Security Engineering Center
Attn: Code 042
3801 Nebraska Avenue, N.W.
Washington, DC 20390

14. Director 1
National Security Agency
Attn: R6
Fort George G. Meade, MD 20755